POLITECNICO DI TORINO

Dottorato in Ingegneria Elettronica e delle Comunicazioni

Tesi di Dottorato

Analytical and Experimental Methods for the Characterization of Field Propagation in Non-Standard Conditions

Relatore:
Prof. Daniele Trinchero

Candidata:
Alessandra Carta

Marzo 2013

Summary

The electromagnetic propagation is totally and fully assessed in free space, in standard working conditions. However there exists peculiar propagation environments in which the propagation has not been studied but in which it could be fully exploited in order to assess specific needs or to provide new sensing tools.

In particular the research activity describes in this thesis has been devoted to the study of the propagation in non-standard conditions.

The first activity has been concentrated on the evaluation of the electromagnetic field in proximity of the antenna, whereas in literature the field is considered just in the so-called Fraunhofer region. The particular interest in the study of the near field propagation is mainly due to health safety purposes. In fact the increasing deployment of antennas for mobile coverage in proximity of living environment has made necessary the study and the monitoring of the electromagnetic field also in the region close to the antenna. Such region is not typically studied and the instruments provided are poorly performing or too costly. The objective of the first presented activity has been the development of a fast, reliable and conservative method for the evaluation of the electromagnetic field generated by a generic antenna in its near-field region. The method has been developed and tested by means of Matlab and compared with full-wave numerical solutions and real measurements.

The second and third activity are both related to the retrieval of snow characteristics for monitoring and sensing purposes and involve therefore the

study of the propagation into the snow. In particular, the second activity is devoted to the realization of an electromagnetic sensor able to monitor the snow status and identify possible avalanche hazard. Avalanches, in fact, can be determined by many factors, among which one of the most important is the amount of liquid water content present into the snow pack. Such content is strongly related to the dielectric characteristics of the snow and it can be therefore assessed by means of electromagnetic sensors. The model of the device has been realized by means of HFSS from Ansoft and it has been tested in different configurations and shape.

The third activity has been performed during the period of internship at Universidad de Zaragoza and it was devoted to the feasibility study of a snow monitoring system based on commercial GPS devices. The target of the proposed system is the Snow Water Equivalent (SWE) that is the amount of water that would result from the melting of the snow pack. Such parameter is very important for water resource exploitation, hydrological studies and management of water supply. The system is based on the deployment of commercial GPS receivers both over and behind the snow cover in order to assess its parameters and retrieve also differential information. The system has been tested in a controlled environment in order to verify the sensitivity of the GPS signal to the variation of the surrounding dielectric.

Contents

Chapter 1

Near field characterization

1.1 Introduction and available methods

In the recent years the rapid and growing diffusion of wireless communication systems such as cellular mobile networks has raised the concern for the possible detrimental effects of their generated electromagnetic fields on the human health. In particular a special attention has been dedicated to the evaluation of the effect of the field generated by mobile phones' antennas on the human head. However the population is not exposed just the radiation of their personal diveces but also to the radiation generated by radio base stations. In 1993 UK Government started establishing guidelines for regulating the maximum levels of exposure to RF radiation emitted from mobile phones, base stations and other sources based on the studies of the National Radiological Protection Board (NRPB). In 1998 the International Commission on Non-Ionizing Radiation Protection (ICNIRP) published its own guidelines. The ICNIRP guidelines for the public have been incorporated in a European Council Recommendation (1999), which has been agreed in principle by all countries in the European Union (EU), including the UK. Both the NRPB and ICNIRP guidelines are based on the need to avoid known adverse health effects, [1]. Nowadays many Health and Safety Governamental Bodies have started requiring an a-priori quantification of the electromagnetic field gen-

erated by broadcasting and mobile installations. In general such studies are perfomed in "'standard"' conditions, in the so-called Fraunhofer region of the antenna, [2]. In such region the propagation is spherical and the antenna gain does not depend on the distance from the antenna. As a consequence, all the information about base station antennas provided by manufactures are referred to this region. However the deployment of antennas closer to living environment (microcell, especially in airports, shopping malls, where the amount of traffic is huge) has raised the need of evaluating the field in the proximity of the antenna. The typical methods are based on far-field simulations in conjunction with ray-tracing techniques that take into account the presences of obstacles. However such methods are based on the use of the antenna radiation pattern and they can therefore be used just in the Fraunhofer region of the antenna and they cannot be applicable in the surroundings of the antenna. In fact the used of standard Fraunhofer radiation pattern for the evaluation of the field in the proximity of the antenna leads to a sensible underestimation of the field itself, as reported in the following figures:

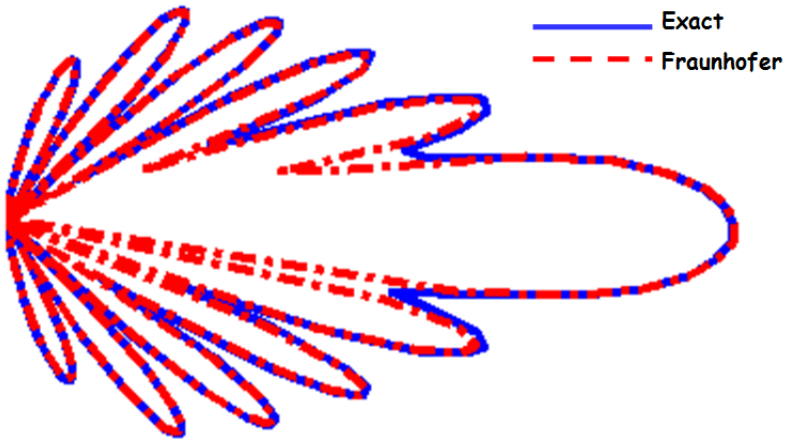

Figure 1.1: Radiation pattern of a collinear array working at 3.5 GHz at 4 meters from the antenna

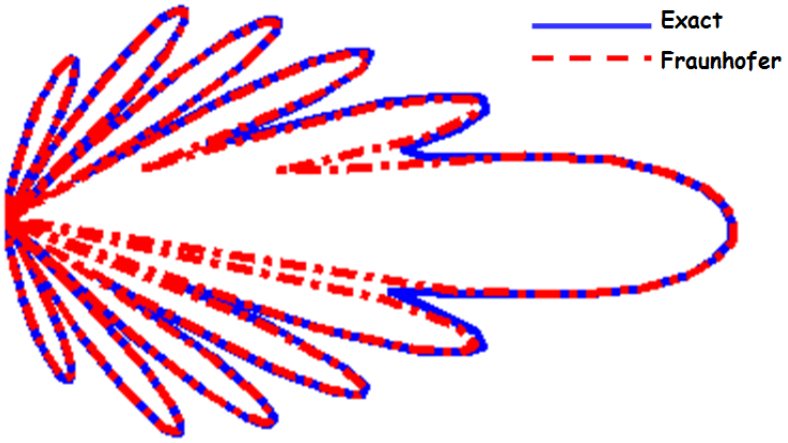

Figure 1.2: Radiation pattern of a collinear array working at 3.5 GHz at 2 meters from the antenna

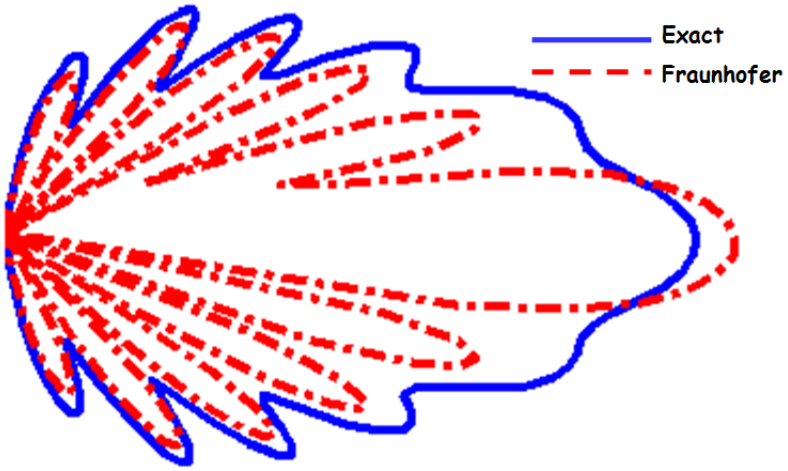

Figure 1.3: Radiation pattern of a collinear array working at 3.5 GHz at 1 meter from the antenna

It is possible to observe that the underestimation introduced the Fraunhofer radiation pattern increases as the distance from the antenna reduces, as clearly shown in 1.3. The Fresnel region is characterized by negligible reactive effects; however it is not possible to define a radiation pattern, since the angular distribution of the radiated field varies with the distance from the centre of the antenna [4], [5].

The other available techniques are based on full-wave numerical methods,

[3]. The most used ones are: the Method of Moments (MoM), the Finite Difference Time Domain and hybrid ones. All the reported techniques provide accurate results but they are characterized by some important drawbacks. First, they all require a precise characterization of the antenna, both electrical and geometrical, which it is not typically available. Moreover such methods are generally very demanding in computational terms.

1.2 Proposed method

The main idea behind the proposed method is to fully exploit the available information provided by antenna manufactures to develop a technique simple, reliable and fast. The objective is also to guarantee an estimation of the field always conservative in order to assure the respect of the exposure thresholds. The proposed approach does not require any knowledge about the current distribution on the antenna and any information about the geometrical configuration of the antenna.

The proposed method estimates the field in the region close to the antenna using the standard Fraunhofer radiation patterns, corrected by means of properly selected analytical Additive Incremental Terms (AIT), [10], [11], [12]. Such terms introduce a dependency not only on the direction of observation but also on the direction of the antenna.

The radiated power density S of a generic antenna in the Fresnel region with the proposed approach has the following expression:

$$S(r,\theta,\varphi) \approx (S_{FR})_{max} f(d_H(\phi) + \delta_H(r,\phi), d_V(\theta) + \delta_V(r,\theta)) \tag{1.1}$$

where S_{FR} is the power density in the direction of maximum radiation, d_H is the normalized pattern in the H-plane, d_V is the normalized pattern in the V-plane, f is a function that combines the two cuts. The AIT δ_H and δ_V are estimated independently, separately calculating in the H-plane and in the V-plane the expressions of the two coefficients Δ_{PH} and Δ_{PV}. This

terms represent the difference between the exact expression of the radiated power density S and its approximation S_{FR}, obtained by a pure Fraunhofer approach. The expressions in the H-plane and V-plane are the following:

$$S(r, \varphi) = S_{FR} + \Delta_{PH} \tag{1.2}$$

$$S(r, \theta) = S_{FR} + \Delta_{PV} \tag{1.3}$$

The evaluate the Δ terms a wire antenna with uniform current distribution is initially considered:

$$\underline{J_e}(\underline{r'}) = I_0 e^{-j2\pi t_M \frac{z}{\lambda}} \delta(x)\delta(y)\hat{z}, -D/2 \leq z \leq D/2 \tag{1.4}$$

The application to such source is trivial but useful in order to introduce its general form that can applied to more complex radiators. The integral representation of the field is expressed according to [8] in the following form:

$$\underline{E_0}(\underline{r}) = -j\frac{Z_0}{2r\lambda}(\hat{\theta}\hat{\theta} + \hat{\varphi}\hat{\varphi}) \int e^{-2j\pi \frac{|\underline{r}-\underline{r'}|}{\lambda}} \underline{J_e}(\underline{r'})\, d\underline{r'} \tag{1.5}$$

where Z_0 is the free space impedance, λ the wavelength, $and\underline{r}$ is the observation point in spherical coordinates. Such expression is valid at a distance r larger than $c_1\lambda$ and $c_2 D$, where D is the maximum dimension of the antenna and $c_1$and c_2 two scalar coefficients that depend on the chosen accuracy.

The terms Δ_{PX} are obtained with a rigorous estimation, for the known current distribution 1.4, of the difference between progressively more accurate representations of 1.5.

By applying a standard approach described in literature, [8], and truncating the Taylor expansion of the exponential component phase of the integrand in 1.5:

$$|\underline{r} - \underline{r'}| = r(1 - t\alpha + \frac{1}{2}(1-t^2)\alpha^2 + \frac{t}{2}(1-t^2)\alpha^3 + O(\alpha^4)) \tag{1.6}$$

($\alpha = r'/r$ and $t = \hat{r}\hat{r'}$) to the its second and third terms, 1.5 results into the two following expressions:

$$\underline{E_{2t}}(\underline{r}) = \underline{E_0}(\underline{r}) D sinc(\frac{D(t - t_M)}{\lambda}) \tag{1.7}$$

$$\underline{E_{3t}}(\underline{r}) = \underline{E_0}\sqrt{\frac{r\lambda}{2(1-t^2)}}[C(\xi_2) - C(\xi_1) + j(S(\xi_1) - S(\xi_2))] exp(j\frac{\pi r}{\lambda}\frac{(t-t_M)^2}{1-t^2}) \tag{1.8}$$

where:

$$\underline{E_0}(\underline{(r)}) = -jI_0\frac{Z_0}{2r\lambda} sin\theta e^{\frac{-j2\pi r}{\lambda}}\hat{\theta} \tag{1.9}$$

$$C(x) = \int_0^x cos(\eta^2\pi/2)\, d\eta \tag{1.10}$$

$$S(x) = \int_0^x sin(\eta^2\pi/2)\, d\eta \tag{1.11}$$

$$\xi_m = (-1)^m\sqrt{\frac{2r}{\lambda(1-t^2)}}(\frac{D}{2r}(1-t^2) - (-1)^m(t - t_M))(m = 1, 2) \tag{1.12}$$

The expressions reported in 1.7 and 1.8 represent the standard expression of the field respectively in the Fraunhofer and Fresnel region of the antenna. Therefore, the Δ_{PX} terms can be expressed as:

$$\Delta_{PV}(r, \theta) = \left|\underline{E_{3t}}(\underline{r})\right|^2/Z_0 - \left|\underline{E_{2t}}(\underline{r})\right|^2/Z_0 \tag{1.13}$$

$$\Delta_{PH}(r, \varphi) = \left|\underline{E_{3t}}(\underline{r})\right|^2/Z_0 - \left|\underline{E_{2t}}(\underline{r})\right|^2/Z_0 \tag{1.14}$$

In the case of an antenna with a generic current distribution it is not possible to compute the Δ_{PX} terms analytically, but they can be calculated

in an approximate form by 1.13 and 1.14. In such hypothesis, the the parameters of 1.4 are computed by imposing that the main-lobe direction and the half-power beamwidth are equal to the corresponding parameters θ_M and θ_{3dB} of the antenna. Moreover, the current amplitude I_0 is determined by imposing that the field 1.7 is equal, in the main-lobe direction, to the field radiated by the antenna:

$$\begin{cases} t_M = cos(\theta_M) \\ D \approx 0.443\lambda / \left| cos(\theta_M + \theta_{3dB}/2) - t_M \right| \\ I_0 = \lambda / D \sqrt{PG_{MAX}/(\pi Z_0)} \end{cases} \tag{1.15}$$

where G_{MAX} is the maximum gain and P is the feeding power.

The method has been tested in order to assess the error introduced by the approximation using as reference antenna the GSM Kathrein antenna 730691 which main characteristics are resumed in the following table:

frequency	900 MHz
polarization	vertical
gain	17.1 dBi
vertical HPBW	9°
horizontal HPBW	65°
height/width/depth	1920/245/95 mm

Table 1.1: Kathrein 730691 GSM antenna main features

The antenna has been modeled and simulated by means of NEC-2 considering an additional electrical tilt of 10°.

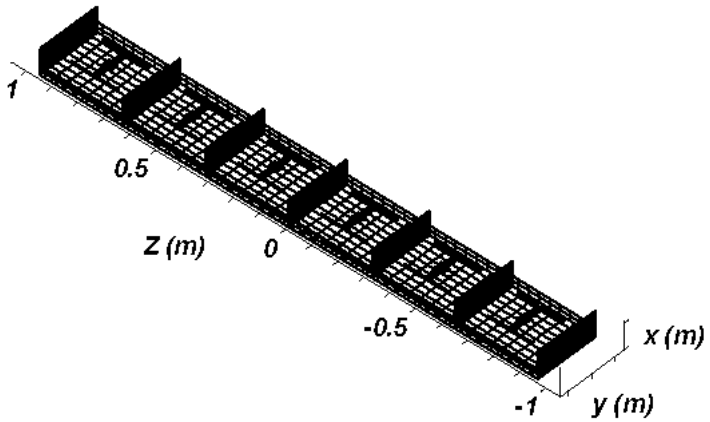

Figure 1.4: NEC model of the Kathrein 730691 antenna

The introduction of the correction term Δ_{PV} is reported in the following figure:

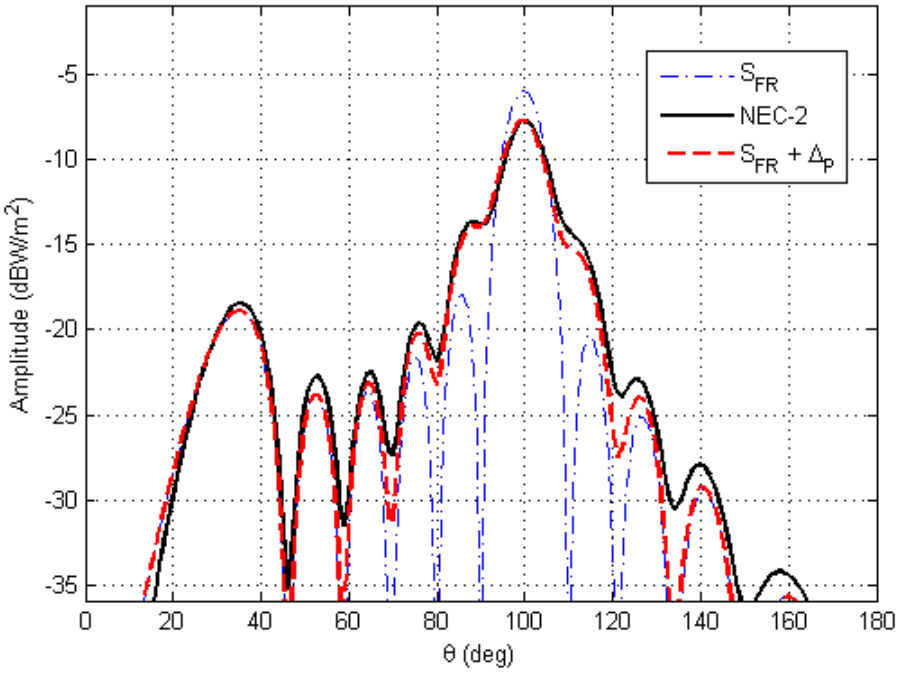

Figure 1.5: Power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W

The reported graph shows a clear underestimation of the secondary lobes,

that becomes more evident performing the simulation closer to the antenna. Therefore the correction term Δ_P has been further corrected with another correction term Δ'_P, in order to avoid any underestimation of the field.

By analyzing the angular dependencies of Δ_P, it has been shown that such parameters generates a power density reduction Δ_0 in the main lobe which is mainly redistributed in correspondence of the first nulls of the Fraunhofer radiation pattern (Δ_1 and Δ_2), 1.6.

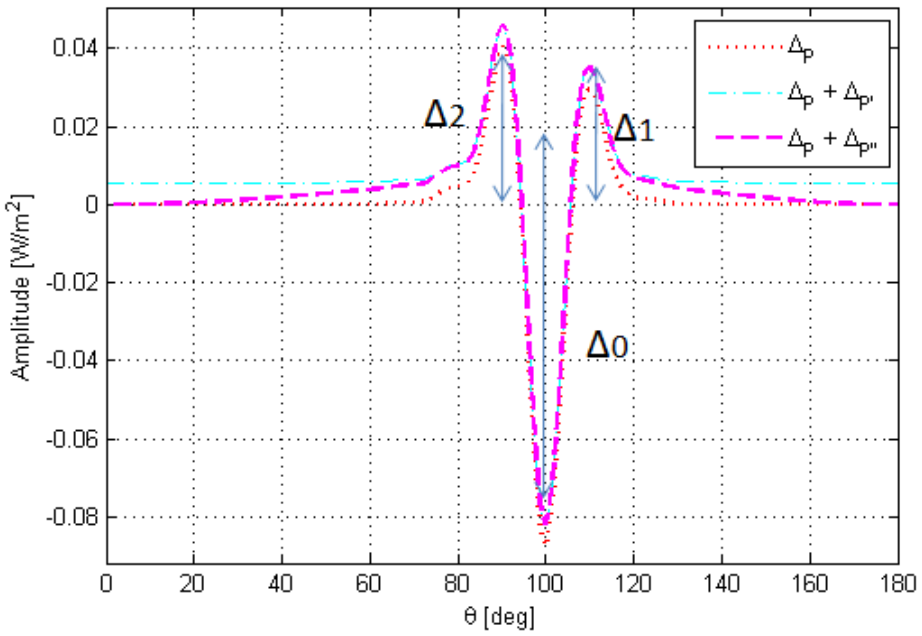

Figure 1.6: Additive terms Δ_P,Δ'_P and Δ''_P

A first proposal for a definition of Δ'_{PV} is to keep it constant and to impose that the quantity $(\Delta_1 + Delta'_P) + (\Delta_2 + \Delta'_P)$ is equal to $(\Delta_0 - \Delta'_P)$. This leads to $\Delta'_P = (\Delta_0 - (\Delta_1 + \Delta_2))/3$. It is possible to observe in 1.6 that such correction term introduces an excessive overestimation applied to secondary lobes. For this reason, Δ'_P is weighted by an appropriate angular-dependent function that takes into account the typical sin^2 behavior of the

Fresnel region:

$$\Delta''_{PV} = \Delta'_P sin^2(\tau(\theta, \theta_M)) \tag{1.16}$$

where:

$$\tau(\theta, \theta_M) = \begin{cases} (\pi/2)(\theta/\theta_M)\theta \leq \theta_M \\ (\theta/2)(1 - (\theta - \theta_M)/(\pi - \theta_M))\theta > \theta_M \end{cases} \tag{1.17}$$

guarantees that $\Delta''_{PV}(r, \theta = 0, \pi) = 0$ and $\Delta''_{PV}(r, \theta = \theta_M) = \Delta'_P$.

The result achieved by the application of the just mentioned correction coefficient is reported in the following figure (magenta line):

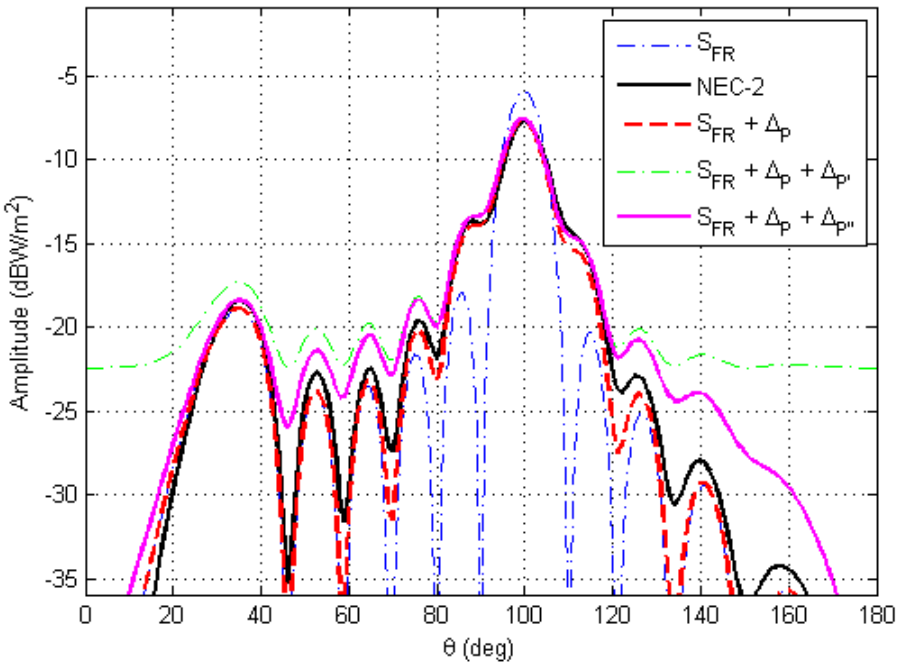

Figure 1.7: Power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W

Finally the AIT in the vertical plane takes the form:

$$\delta_V(r, \theta) = \frac{\Delta_{PV}(r, \theta) + \Delta''_{PV}(r, \theta)}{max S_{FR}(r, \theta)} \tag{1.18}$$

An analogous expression can be computed for δ_H.

The achieved performance of the method is satisfactory, as it leads to a good and always conservative estimation of the field. However the method, that proved to work very well when dealing with antennas characterized by uniform current distribution, when applied to non-uniform current distribution exhibits an underestimation in correspondence of the main lobe, as reported in 1.8 and 1.9.

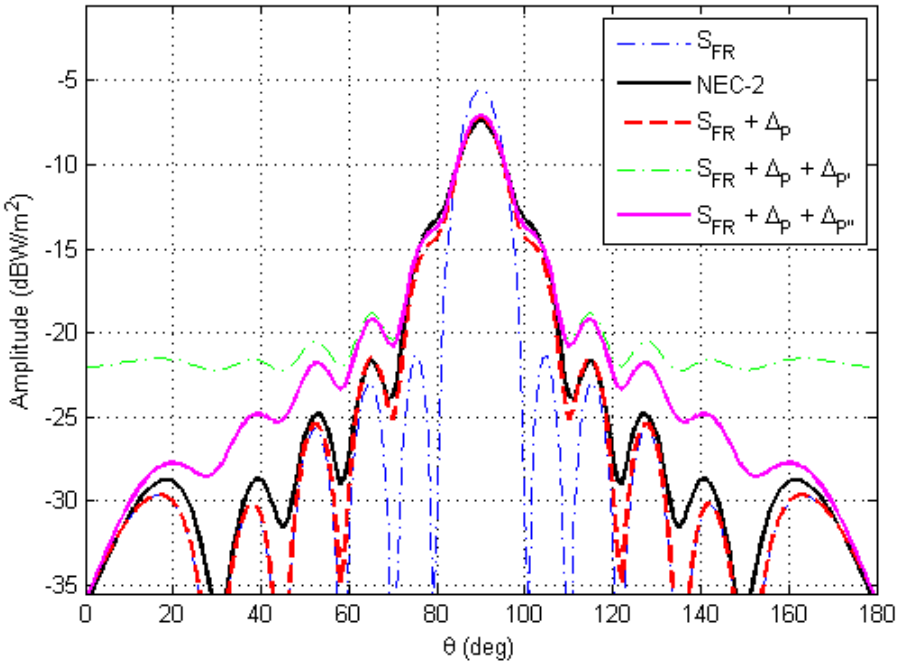

Figure 1.8: Power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, Tschebyscheff current distribution

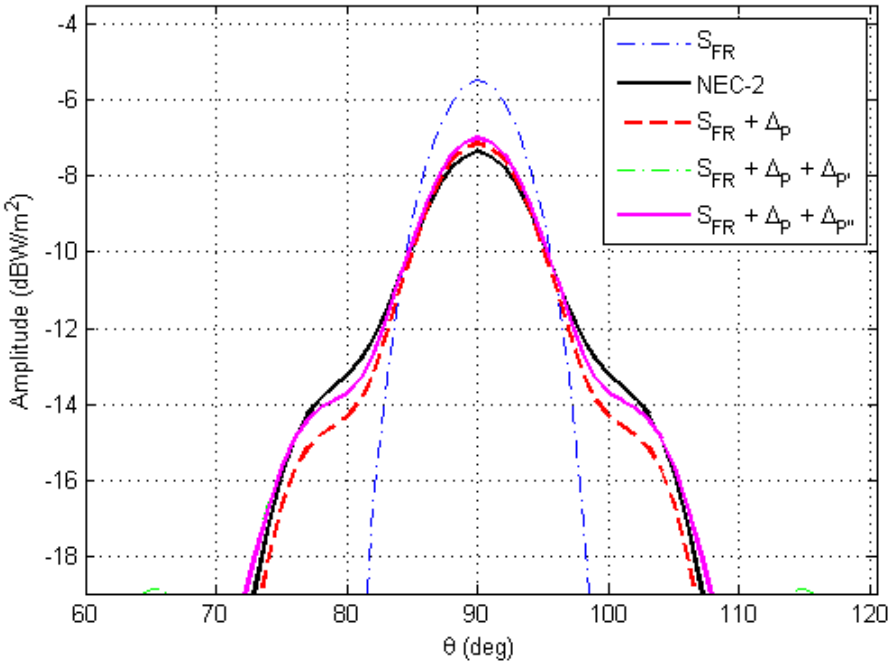

Figure 1.9: Detail of the main lobe of the power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, Tschebyscheff current distribution

The main purpose of the method is to always guarantee overestimation of the field, in order to represent a reliable instrument for the monitoring of human exposure. Therefore the method has been further modified in order to be suitable to represent any generic non-uniform current distribution.

First, the method has been improved by modeling the current as a piece-wise function over each element of the antenna, [13], [14].

$$\underline{E_{2t}}(\underline{r}) = \sum_{i=1}^{N} \underline{E_{2t}^{i}}(\underline{r}) \tag{1.19}$$

$$\underline{E_{3t}}(\underline{r}) = \sum_{i=1}^{N} \underline{E_{3t}^{i}}(\underline{r}) \tag{1.20}$$

where N is the number of elements of the array.

However the geometry of the antenna and in particular the number of

elements of the antenna are not a-priori known. The analysis, by means of simulations, of the relationship between the total number of elements of the antenna and some parameters of the radiation pattern, as the Half Power Beam Width (HPBW), the amplitude of the lobe at -10 dB and the First Null Beam Width (FNBW)show that the main lobe properties do not vary with respect to the total number of elements. Thus it is possible to represent an antenna of a given length with a different number of elements spaced in a proper way, 1.10, 1.11 and 1.12.

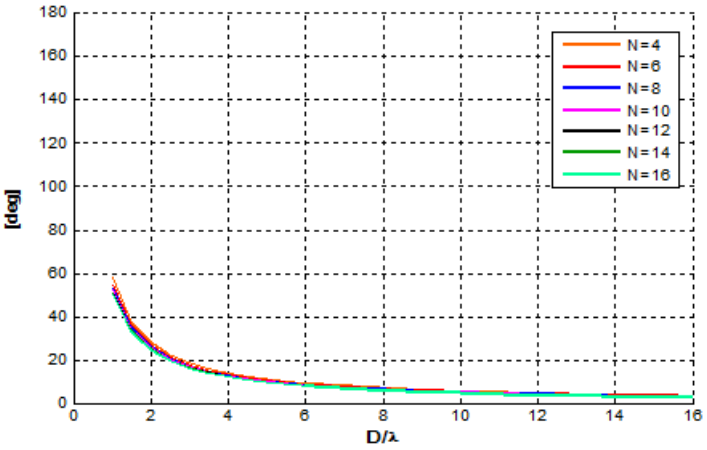

Figure 1.10: Half Power Beam Width computed for a Tschebyscheff array with different number of elements and spacings

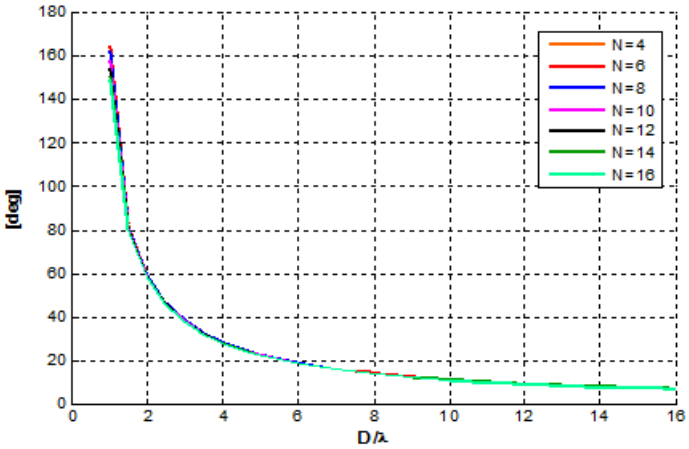

Figure 1.11: Amplitude of the lobe at -10 dB computed for a Tschebyscheff array with different number of elements and spacings

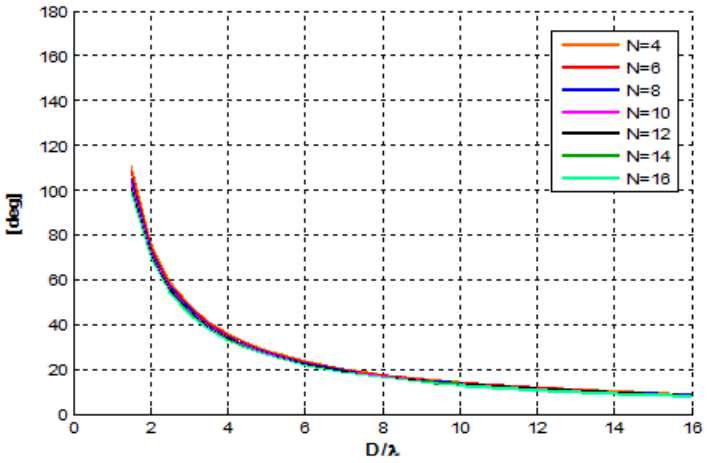

Figure 1.12: First Null Beam Width computed for a Tschebyscheff array with different number of elements and spacings

The corresponding excitation coefficients to apply to the elements of the antenna can be inferred from its array factor. It can be obtained from the standard Fraunhofer radiation pattern provided my manufacturers, deembedding the contribution of the single element and of the shield (reasonably considered present). In the case of Tschebyscheff array the array factor gives also the information on the side lobe level ratio R_0. Moreover the current over each element has been modeled as a sinusoidal function, making it more realistic, 1.21, [15], [16].

$$I_0 = I_0^i cos(k_0(z - z_{Mi})) \tag{1.21}$$

As a result the two summations of 1.19 and 1.20 have the following expressions:

$$\begin{aligned}\underline{E}_{2t}^i(\underline{r}) = \underline{E}_0(\underline{r}) z_{Di}[&e^{-jk_0 z_{Mi}} e^{jk_0 z_{Mi}(t-t_M+1)} sinc(2\frac{z_{Di}}{\lambda}(t - t_M + 1)) + \\ &+ e^{-jk_0 z_{Mi}} e^{jk_0 z_{Mi}(t-t_M-1)} sinc(2\frac{z_{Di}}{\lambda}(t - t_M - 1))]\end{aligned} \tag{1.22}$$

$$\underline{\underline{E}}^i_{3t}(\underline{r}) = \underline{\underline{E}}_0\sqrt{\frac{r\lambda}{2(1-t^2)}}[e^{-jk_0 z_{Mi}}(C(\xi_{21}) - C(\xi_{11}) + j(S(\xi_{21}) - S(\xi_{11})))$$
$$e^{jk_0\frac{r(t-t_M+1)^2}{2(1-t^2)}} +$$
$$+ e^{jk_0 z_{Mi}}(C(\xi_{22}) - C(\xi_{12}) + j(S(\xi_{22}) - S(\xi_{12})))$$
$$e^{jk_0\frac{r(t-t_M-1)^2}{2(1-t^2)}}] \tag{1.23}$$

where:

$$\begin{aligned}
\xi_{n1} &= \sqrt{\frac{2r}{\lambda(1-t^2)}}\left(\frac{\frac{z_{ni}}{r}(1-t^2-\frac{1}{2}\sigma) - (t - t_M - \frac{1}{2}t\sigma) + 1)}{\sqrt{(1-t^2)}}\right) \\
\xi_{n2} &= \sqrt{\frac{2r}{\lambda(1-t^2)}}\left(\frac{\frac{z_{ni}}{r}(1-t^2-\frac{1}{2}\sigma) - (t - t_M - \frac{1}{2}t\sigma) - 1)}{\sqrt{(1-t^2)}}\right) \\
&(n = 1, 2)
\end{aligned} \tag{1.24}$$

$$z_{Mi} = \frac{z_{1i} + z_{2i}}{2} \tag{1.25}$$

$$z_{Di} = \frac{z_{2i} - z_{1i}}{2} \tag{1.26}$$

and with z_{1i} and z_{2i} equal to the lower and upper limit of each element.

The last adjustment introduced to the method consists into the introduction into the model of the presence of the shield. In fact, base station antennas enhance directivity and front-to-rear separation implementing a metallic shield. A very simple and effective way to model such effect is to consider the alignment of the antenna and the shield as an array of two elements spaced twice the distance between them. The correction term Δ''_{PV} can be consequently updated by the introduction of additional multiplicative terms: AF_{shield} that is a two elements array factor that models the presence of the shield and $F_{current}$ that is a piece-wise constant function, obtained

as the mask of the array factor of the antenna, that properly describes the current distribution shape of the considered radiator.

$$\Delta''_{PV} = \Delta'_{PV} sin^2(\tau(\theta, \theta_M)) AF^2_{shield} F^2_{current} \tag{1.27}$$

1.3 Results

The proposed method has been designed in Matlab (see appendix A) and validated by comparing its results with simulations performed using the NEC model of the GSM Kathrein 730691 antenna, which features are reported in 1.1. For the purpose of the study, the excitation of the antenna elements has been modified, in order to introduce several array combinations. First the method has been applied to the Tschebycheff current distribution. Moreover some hypothesis has been done in order to apply the method: the single element is a dipole and the distance between the shield and the antenna as been set to . The combination of such information with the known length D of the antenna allows the creation of a set of pairs (N, λ) that can all equally represents in a proper way the considered antenna. The method has been tested with a configuration of 6 elements spaced of λ. The black line represents the result obtained by full-wave simulations performed with NEC2 and the magenta line represent the final estimation given by the development method with all the improvements added to its basic version.

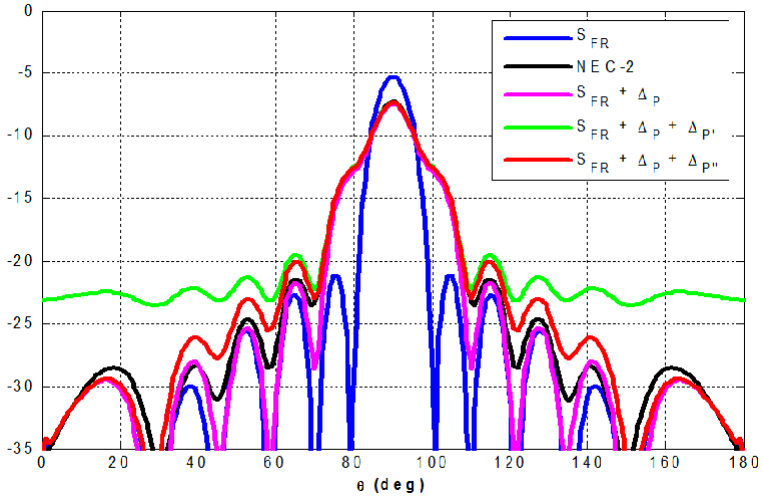

Figure 1.13: Power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, Tschebyscheff current distribution - $R_0 = 15dB$

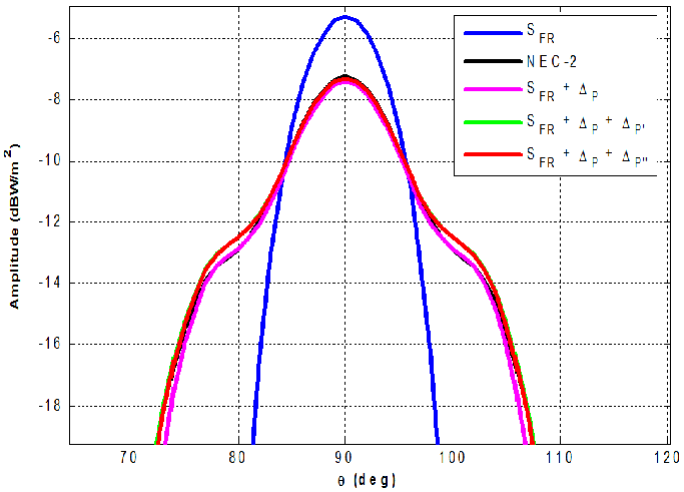

Figure 1.14: Detail of the power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, Tschebyscheff current distribution - $R_0 = 15dB$

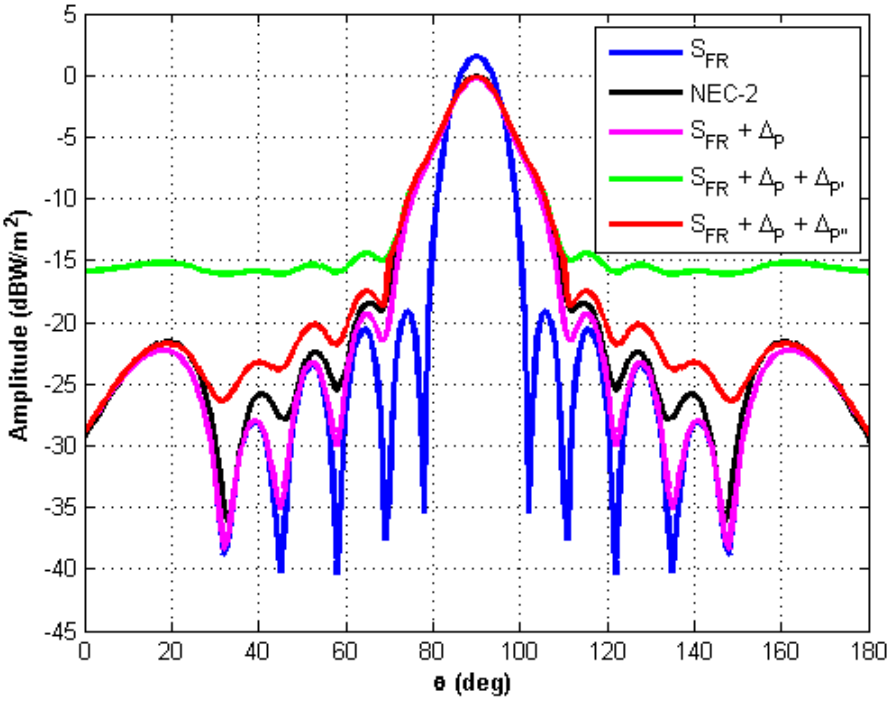

Figure 1.15: Power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, Tschebyscheff current distribution - $R_0 = 20dB$

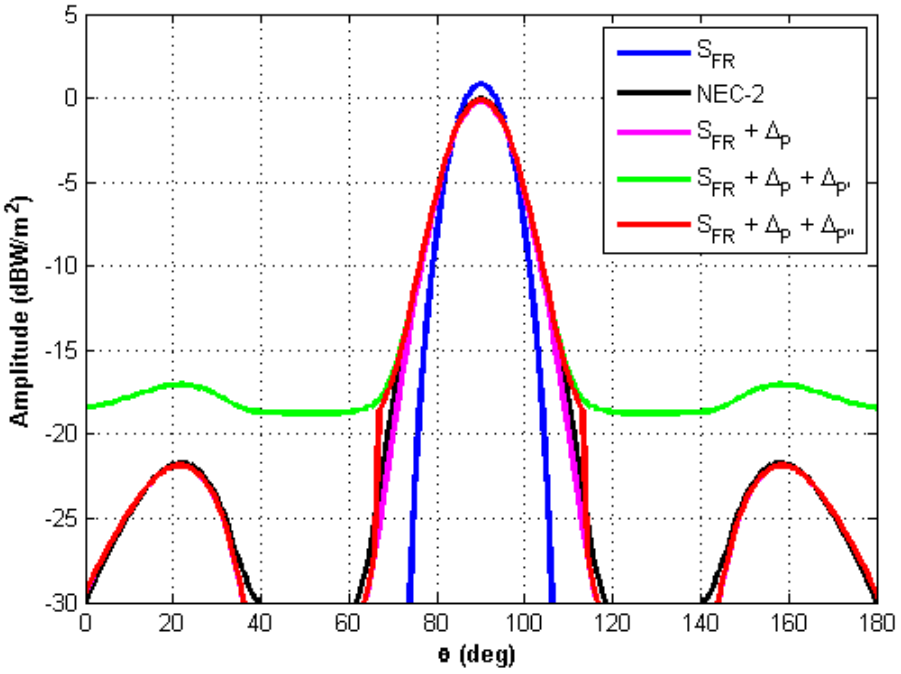

Figure 1.16: Power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, Tschebyscheff current distribution - $R_0 = 40dB$

In all the reported graphs the obtained estimation is really close to the NEC2 simulation, both in the evaluation of the main lobe, and of the secondary lobes, guaranteeing in the same time always the overestimation of the field. In the following are reported same graphs that compares the performance of the method in its two different version. It is possible to appreciate how the refinements introduced improved the whole performance of the method, both concerning the estimation of the main and secondary lobes.

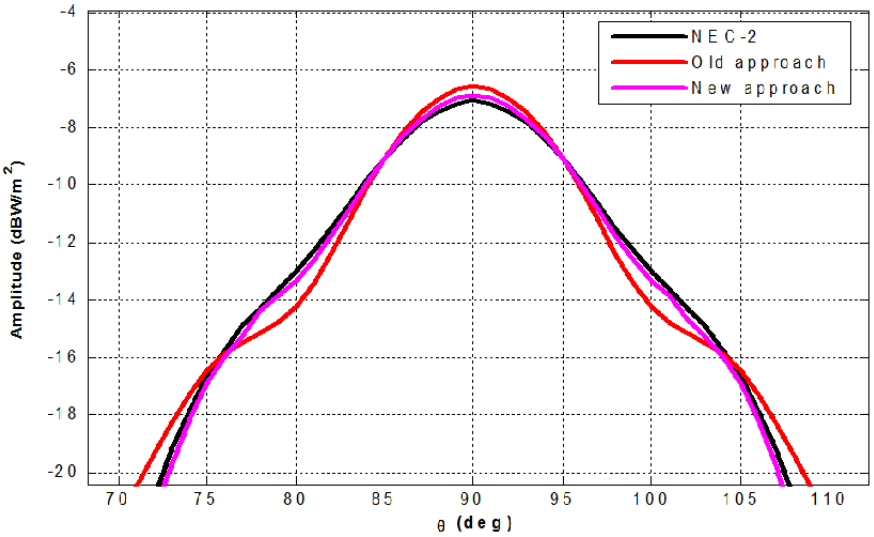

Figure 1.17: Detailed comparison (main lobe) between old and new method of the power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, Tschebyscheff current distribution - $R_0 = 20dB$

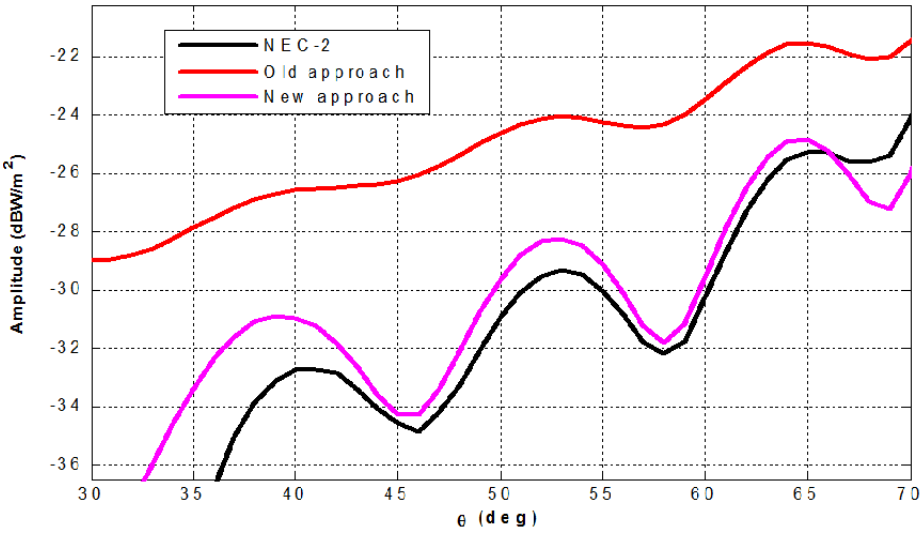

Figure 1.18: Detailed comparison (side lobes) between old and new method of the power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, Tschebyscheff current distribution - $R_0 = 20dB$

The good results achieved by the new modified version of the method made possible to apply it also to the standard uniform current array, achieving also in this case improved performance with respect to the basic version.

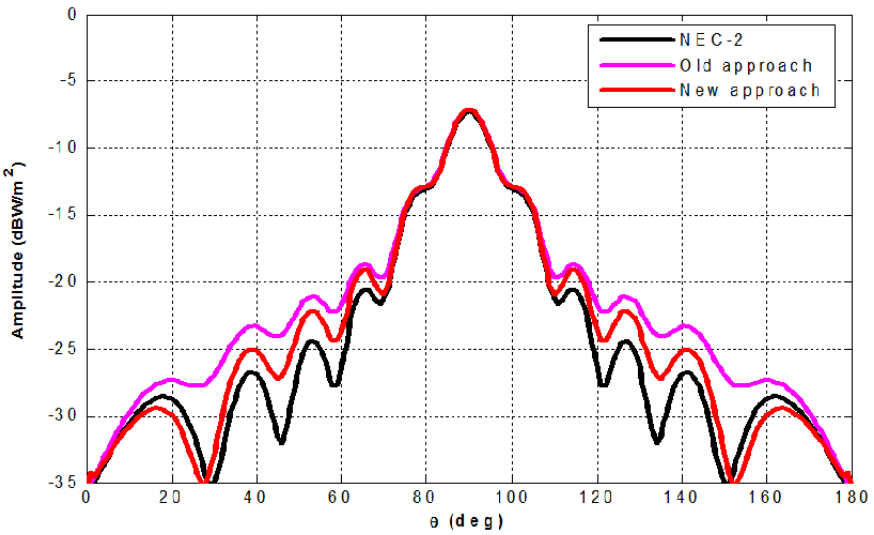

Figure 1.19: Comparison between old and new method of the power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, uniform current distribution

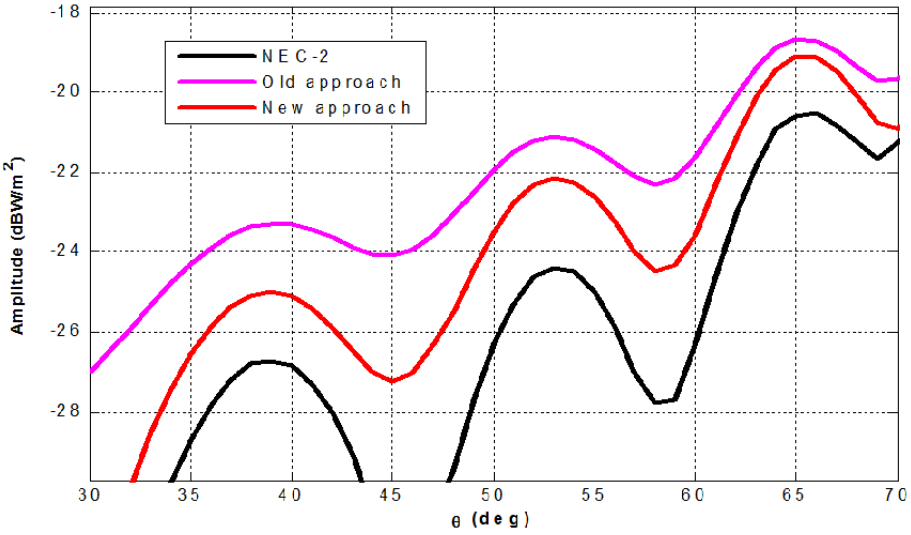

Figure 1.20: Detailed comparison (side lobes) between old and new method of the power density radiated in the vertical plane at a distance of 4 m from the center of a Kathrein antenna 730691, fed with 1 W, uniform current distribution

1.4 Conclusion

The development method represents an useful and easy-to-use instrument for the evaluation of the field in the proximity of the antenna. The proposed approach does not require any a priori knowledge of the antenna geometry and of its electrical configuration and making use of the standard information provided by manufactures is able to achieve a reliable and efficient estimation of the field. Moreover the method is fast and always conservative and it can be apply to any antenna, with any current distribution, without losing in accuracy.

Chapter 2

Snow characteristics and sensing techniques

2.1 Snow characteristics

Snow is a complex medium, highly porous, organized in a microstructure constituted by a continuous ice structure and a continuously connected pore space. The snow is very sensitive to temperature variations and therefore it is subject to a continuous process of transformation, known as metamorphism. Moreover new precipitations and the action of the wind induce additional change to the snow, generating different distinct layers into the snowpack. Such layers are characterized by different density, snow hardness, liquid content, snow temperature and impurities, which translates into a high degree of variability of the snowpack, both in the vertical and horizontal direction. However seasonal snow classification [17] has been standardized by a periodic international publication that describes the most important snow features and the way of measuring and evaluating them. Among them the grain shape, snow density, hardness, liquid water content and layer thickness are the most relevant ones. Snow is a granular material and the grains could have many different shapes and sizes. The grain shape can be determined directly on the field by means of a crystal card and a magnifying glass, or with a stereo-microscope if a specialized work is needed. The grain size instead

can be evaluated just placing a snow sample on a plate with a millimeter grid. In this way it is possible to measure both the average and maximum size of the grains by comparison with the spacings of the grid. However the optical equivalent grain size (OGS) is preferred as it represents the electromagnetic characteristics of the snow in order to be considered for remote sensing applications. Note that the grain size should be considered as a property of the snow layer and not of the grain shapes. Another important property is the snow density, typically evaluated in mass per unit volume and determined by weighting snow of a known volume. However the density evaluation can be performed also by means of electromagnetic techniques and devices. The snow density is subject to variations due to the metamorphism and it is one of the main parameter to take into account for the evaluation of the layers state. The snow hardness is instead the resistance of the snow to the penetration of an object. The result obtained testing the snow hardness depends both on the operator and the used instrument that should always be specified. The most common tests are the hand test (performed using different objects with decreasing area) and the rammsonde test (quasi-objective measurement in newtons). A parameter that has a strongly impact for the evaluation of the snow resistance and the cohesion among the different layers is the liquid water content, namely the total amount of water within the snowpack which is in the liquid phase. Liquid water into the snowpack can be due to rain and melting and it is typically measured in volume or mass fraction. Several techniques can be used to determine such parameter: freezing calorimetry, alcohol calorimetry, dilution method and dielectric measurements. The state of the snowpack can be evaluated also taking into account the snow layer thickness, typically measured vertically in centimeters. The evaluation of the snow thickness can be further extended considering the total height of the snowpack (the distance from the snow surface to the base), also known as snow depth. The measure of the snow depth should be always be referenced to a specific location and a given time. Snow depth can be evaluated by

means of ultrasonic sensors and also by new remote sensing techniques.

2.1.1 Snow Water Equivalent

A very important parameter is the snow water equivalent (SWE) that is the depth of water that would result if the snow melted completely. In fact the evaluation of the SWE assumes a fundamental role for many disciplines, including water resources research, hydrological studies and for the management of water supply and flood water budget estimation. The SWE can be obtained as the product of the snow depth and the vertically integrated density and it is typically expressed in kilograms or liters per square meter. Many available techniques make possible to infer both snow depth and snow density, allowing the computation of SWE. However it can be simply measured by weighting samples of known cross section. All the mentioned parameters are necessary to perform a full description of the snow status but fortunately most of them can be determined or at least estimated by exploiting different available technologies.

2.2 Techniques for measuring snow depth

2.2.1 Ultrasound

Ultrasound represents one of the techniques used to the automated retrieval of the snow depth. Such method is based on the use of longitudinal disturbances that propagate through a medium; the velocity of ultrasound in the medium depends on the density and elasticity of the material. In particular, the Ultrasonic Snow Depth Sensors (USDS) emit an ultrasonic pulse at 50 kHz and measure the time that it is needed to return to the sensor itself. The emitted pulse consists of a cone of about 20 degrees in which the measure is performed. Such cone should be completely free from obstacles, included trees, wires and interferences of other sensors, as reported in the following figure.

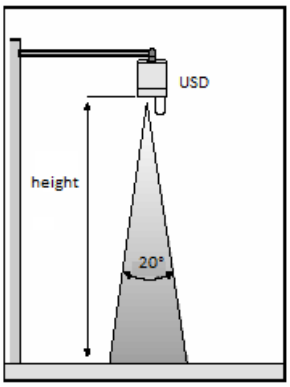

Figure 2.1: Configuration of the ultrasound depth sensors measurements

The measure is corrected for air temperature using a thermocouple attached to the underside of the sensor. The time is converted to the distance via an internal algorithm taking into account also the height of the sensor from the ground. Each measurement cycle consists of two measurements, in order to compare them and check their reliability. In fact, if the difference between the two measurements is less than 1 cm, the value is saved and outputted. Otherwise, if the difference is greater than 1 cm, the oldest measurement is discarded and another one is performed. Such iterative process lasts to a maximum of 10 attempts. If it is not possible to record a valid measure, the USDS send no echo or a zero value. The record of a measurement equal to zero gives as final reading a value equal to the total height of the sensor from the ground and represents one of the most common erroneous readings that the user can observe.

Such kind of devices is typically able to measure till a depth of 6 meters, with a resolution between 1 and 3 mm, depending on the manufacturer. The measurement obtained by means of USDS is typically highly correlated to the manual measurement (errors around 0.%for depth between 0 and 3 m and 15% for higher depths). However the ultrasonic techniques can be affected by many causes of errors. First of all the sensor performance could be affected by misplacement; in fact the sensor should be perpendicular to the target object in order to get an accurate result. Other causes of errors

could be that the target is small and reflects little sound, the target surface is rough and uneven, the target is a poor sound reflector, the transducer is obstructed by snow or ice and strong winds can blow the echo out from under the sensor. Moreover heavy snowfall can cause an attenuation of the sound pulse as the USDS needs a clear path beneath the transducer in order to send and receive a quality measurement. In conclusion to obtain reliable and consistent results by means of ultrasonic sensors it is necessary to select properly the siting. The structure of the sensor should be strong enough in order to maintain the sensor perpendicular to the snow surface. Moreover, as the snow is characterized by a high spatial variability, it would be better to place more than one USDS in order to get a proper picture of the snow depth in the considered site.

2.2.2 RADAR

The use of Frequency Modulated Continuous Wave (FMCW) is one of the possible alternative techniques for the remote monitoring of the snow depth. The FMCW radar generates an input signal, typically ramped from low to high frequency. Such signal is mixed with an incoming signal delayed by the time necessary to it to travel to the ground (or snow), be backscattered and return. The resulting time delay is very short, in the order of 10-8 s, and thus it cannot be easily measured. However the introduced delay translates into a frequency shift between the outgoing and incoming signal easy to measure and proportional to the distance travelled. In fact, to higher frequencies correspond longer travelled distances. The frequency shifts are recorded in the time domain and then converted into the frequency-domain by a fast Fourier transform (FFT). The resulting frequency spectra allow identifying the reflectors. Moreover the distance between two reflectors, for example the snow and the ground can be measured and translated into a distance using the following formula:

$$D = \frac{\Delta f t_s c}{2 B_w \sqrt{\epsilon_s}} \tag{2.1}$$

Where Δf is the frequency distance, t_s is the sweep time of the ramped signal,c is the speed of light, B_w is the bandwidth of the ramped signal and ϵ_s is the dielectric constant of the considered medium. It is relevant to notice that the snow dielectric constant varies according to its density and liquid water content and therefore its value should be determined considering snow samples collected on the measurement field.

The measurement by [24] has been performed in the X-band (8-12 GHz), which gives the best trade-off between resolution and penetration depth and allows the assessment of both wet and dry snow. The deployed antenna was a classical horn antenna placed on a frame on the side of a sled, that house also an operator, a computer and a GPS. The sled performed numerous individual traverse lines to scan the snow surface. The GPS module hosted on the sled has been used to localize the data and record the start and end points of each scan line. In conjunction with the radar measurements, a set of manual measurements has been performed in the same field, in order to provide a coherent reference. The achieved results are similar to that obtained by manual probing data, but they are not equal (around 6 cm of uncertainties over a snow cover depth of 1 meter). The reason of such difference is attributable to the possible incorrect interpretation of the radar data and to uncertainties due to the interaction between the radar signal and the environment. In fact radar snow depth are interpreted measurements, that can be contains large errors if the different layers and reflectors are incorrectly identified. The identifications of peaks generated by the snow is usually simple as its surface is typically smooth compared to the radar wavelength. On the other hand the signal reflected by the ground is not as easy to identify, mainly because of the high attenuation, the volume scattering within the snow and the internal reflections caused by the different snow layers. Other causes of errors affecting the measurements can be ascribed to the movement of the antenna

due to the sliding over an even surface, making the antenna pointing in a direction different with respect to the one where the manual measurement were performed. Moreover the probing makes the measurement in a single point, whereas the radar samples over a finite surface determined by its height over the ground and its beamwidth. In addition, all the snow depth calculations supposed a constant snow density along all the traverse line, but the manual probing demonstrated that in the whole measurement field the density varied substantially. Other authors, [23], used radar as a remote sensing technique for the determination of the snow depth, using airborne radar, or placing the radar on a tramway over the ground and considering also different frequencies. The obtained results show an error of about 20 cm on a depth of 1 meter and are affected by the same sources of uncertainties previously mentioned. In conclusion the described technique can estimate snow depth with an error which is typically less than the 7%. In addition the maximum range is of 2 meters of snow. Moreover the necessity of knowing the dielectric constant of the considered snow pack and its density makes this technique critical.

2.2.3 LIDAR

The Light Detection and Ranging (LIDAR) is an airborne remote sensing technology that seems very promising for the assessment of snow depth. In fact such method provides high spatial point density over large surface extent. The calculation of the snow depth by means of LIDAR data should be performed using two data collections, one with snow-free and one for snow-covered ground. The data are subtracted in order to obtain just the snow depth. More specifically, the LIDAR is a ranging instrument that measures the target distance by calculating the elapsed time between emitted and return lasers signals. The laser signal is typically centered at wavelength $\lambda = 1024nm$. The position of the aircraft platform is determined by means of GPS triangulation, whereas the platform orientation is determined by means of an Inertial Navigation System (INS) link. Once such positions have been

determined is possible to use the time of return of the laser pulse to calculate the 3D locations of the laser points. The point density at ground level is influenced by many factors, including the scan pattern, the scan rate, the scan angle, the swatch width, the pulse rate and the aircraft height.

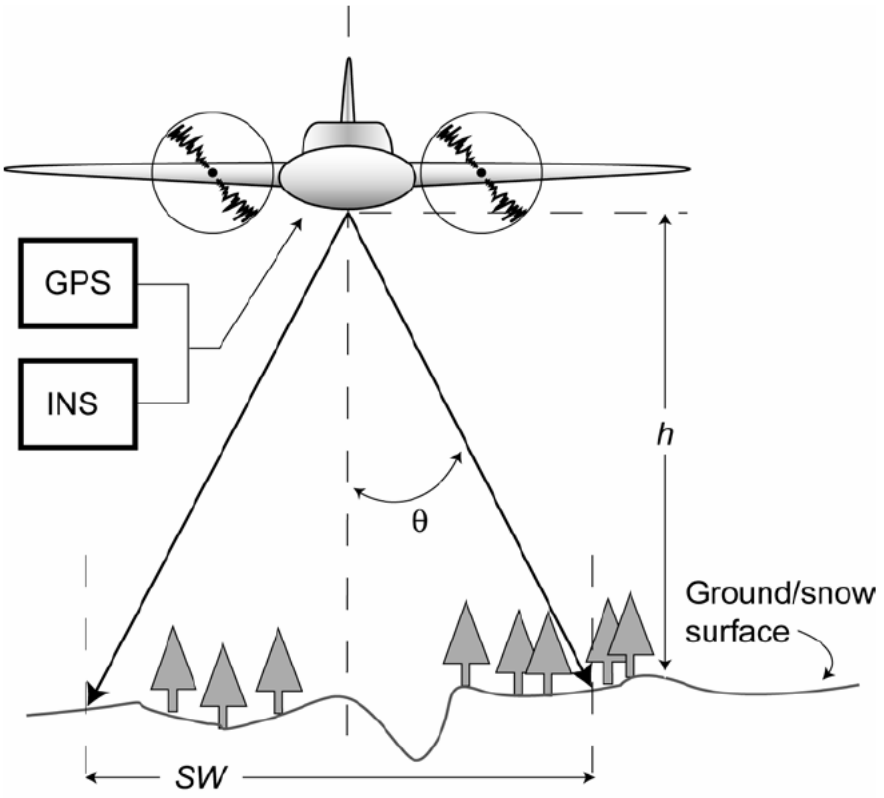

Figure 2.2: Configuration of the LIDAR system

The most common scan patterns are parallel or Z-shaped bidirectional scans. Another used scan pattern is the elliptical one that provides much opportunity for canopy penetration. Scan angle is typically set to a value of $\pm 15^{\circ}$, which is sufficient to guarantee a good level of penetration. The scan rate is the angular velocity of the oscillating mirror that directs the outgoing laser pulse and its typical value is in the range of 30 Hz. However the most important parameter to determine the across-track point distance is the laser pulse rate, which in the modern LIDAR sensors is of 100 kHz, allowing for

very dense laser shot patterns. The swatch width is very important for the mission planning. In fact the data collection cost could be sensibly reduced performing wider swatch that allows a larger coverage with a fewer number of flight strips. However swatch width represents a tradeoff between cost and accuracy and thus should be properly calibrated. The raw data collected by the LIDAR are in the form of points represented by the 3 coordinates (x,y,z). Such data should be filtered in order to assure that all the collected points belong to the same surface. However such filtering activity is mostly performed by means of automatic algorithms monitored manually. Once the filtering has been performed it is possible to subtract the snow-free ground elevation from the snow-covered elevations to obtain the snow depth. However this subtraction cannot be done in the form of point-to-point subtraction, as the likelihood of ground and snow points existing at the exact (x,y) location is quite small. Therefore the data should be converted to a grid dataset by means of interpolation that introduces some errors, minimized thanks to the high spatial resolution of the points. LIDAR snow depth estimation can be affected by different kind of errors due to positioning system, the flight planning, the presence of vegetation and the post-processing of the data. The GPS and INS give the position of the platform and therefore it should carefully verified that the laser range measurements are properly linked to the appropriate positional data. As said previously, the flight planning represents a trade-off between cost and accuracy. In particular, a proper design of the flight avoids the collection of points with poor geometry, especially in presence of slope terrain. The presence of vegetation can obstruct the view of the surface, but this drawback could be overcome simply increasing the pulse rate and decreasing the scan angle. Finally, the post-processing of data can induce a misclassification of points, leading to a non-accurate resulting elevation measurement. Moreover the interaction between the snow and the LIDAR pulses should be further investigated, in order to verify how different kind of snow and snow grains can affect the total final estimation. The hori-

zontal resolution achieved by the use of LIDAR is of the order of 1-2 meters, whereas the snow depth can be retrieved with decimeter-scale accuracy.

2.3 Techniques for measuring Snow Water Equivalent

2.3.1 Cosmic ray neutrons

The remote sensing of SWE can be performed in a non-destructive way taking advantage of techniques based on the absorption of some kind of radiation. The assessment can be done simply by a method of absorption of artificial radiation, as gamma radiation. However such technique can measure a snow-water depth of just 1 meter. In addiction several safety problems arise concerning the use of this kind of radiation. Therefore the estimation of the SWE is typically performed using the attenuation by snow of cosmic ray produced by neutrons. In fact, since the rate of absorption of cosmic ray by snow is much lower than that for gamma rays, it is possible to use such method even for snow covers with a thickness higher than 1 meter water equivalent. Moreover this kind of rays overcomes all the issues related to the radiation hazard. In addition, cosmic rays are equivalent to an infinite plane source coming down from the upper atmosphere and, as a consequence, they are free from scattering effect. The exploitation of this technique has made possible the realization of the so-called cosmic ray snow gauge, developed by [29] and now available on the market, that allows the continuous monitoring of the snow.

Such device consists of a sensor and a recorder, which registers the number of neutrons detected by the sensor. The sensor is a moderated BF3 (Boron Trifluoride) proportional counter with a 2 cm polyethylene tube that has an excellent good stability and long-life and works really well even under severe weather conditions. The sensor is able to detect the attenuation caused to the rays by the passage through the snow. Such attenuation is

Figure 2.3: View of the cosmic ray gauge for the snow water equivalent estimation

related to the amount of liquid water content within the snowpack. The experimental results show a very good agreement (±5% of error) with the manual snow probing, even if this device is affected by some source of errors that should carefully be considered. First of all, the cosmic ray intensity observed on the ground is inversely proportional to the barometric pressure. Therefore a daily variation of barometric pressure of 20 mb corresponds to about a 15% in neutron variations, which respectively translates into 10 cm of water equivalent of snow. As a consequence, it should be necessary to correct the observed neutron count for pressure change, before converting it into SWE. Another additional source of uncertainties is the great variability of the intensity of the primary cosmic radiation from outside the Earth's atmosphere. However this kind of phenomena is accurately monitored by the world-wide network of cosmic ray observatories and the proper corrections could be easily applied. Another source of small errors in the SWE evaluation is related to the albedo neutrons produced by cosmic rays within the soil and/or backscattered from soil to air. The equilibrium condition of this kind of neutrons is influenced by the amount of water into the soil. Therefore when the snow cover is thin and the moisture content of soil is high, the attenuation by these albedo neutrons close to the ground surface can generate an apparent and transient increase of water equivalent of snow. However the impact of this effect on the final result is very small. This technique is currently used in different countries all around the world. For example, the automatic hydrologic information system of the Ebro river basin in Spain

includes fifteen snow meter using cosmic rays placed in relevant point of the basin. In conclusion the use of cosmic rays for assessment of the snow water equivalent is a very reliable technique, which however must be preferred for long term monitoring than for the fine tracing of rapid temporal variation of snow water equivalent. The achieved accuracy on the final measurement stays in the range between 5-15%. However such technique is characterized by high cost, due to the needed infrastructure and instrumentation and to the resulting maintenance costs. Moreover it can give just a reduced spatial resolution, as the device structure allows only few sensing points.

2.4 Techniques for measuring snow electrical properties

Snow is a complex dielectric medium that can be analyzed sensing the variation of its dielectric constant. In fact, as demonstrated by many authors [38], [36] and [37] the dielectric constant of snow can be directly related to the snow density and to its liquid water content. In particular the dependence of the real part and imaginary part of the dielectric constant has been analyzed considering dry and wet snow. Dry snow can be considered, from the electromagnetic point of view, as a heterogeneous medium composed of ice and air. The dielectric constant of ice do not vary in a wide range of frequencies (between 10 MHz and 1000 GHz), and thus the permittivity of dry snow is dominated by the snow density effect. Many models have been developed in order to determine an exact expression to relate permittivity and density,[39], [36], [37] and [38]. The resulting expressions give a linear dependency between the real part of the dielectric constant of dry snow and its density. The imaginary part, instead, is quite small and shows a dependence on density, temperature and frequency. However such dependency can be modeled, in terms of tangent loss, with an equation valid for all temperatures and frequencies. On the other hand, wet snow is a dielectric mixture

of ice, liquid water and air whose behavior depends both on frequency and density. Moreover wet snow is characterized by two different regimes of liquid water saturation: the pendular regime and the funicular regime. The pendular regime is characterized by the presence of a continuous distribution of air in the pore structure and by the presence of isolated inclusions of liquid water. Such regime correspond to a low level of liquid saturation, in general less than 7%. The funicular regime corresponds, instead, to a higher level of liquid saturation in which liquid water is continuously distributed in the whole pore space and the air is trapped into isolated bubbles. The dielectric constant of wet snow in the pendular regime is independent on the snow structure and the behavior seems to be dominated by the water effect. The transition between the pendular and the funicular region represents a critical point for the evaluation of the snow wetness. To sum up, the variation of the dielectric permittivity of dry snow can be almost exclusively related to its density, whereas in wet snow such variation is associated with density and liquid water content. The imaginary part of the dielectric constant in case of wet snow depends on the liquid water volume. The increase in both the real and imaginary part of the dielectric constant of wet snow, with respect to dry snow, is related to the wetness in a similar way. Therefore it is possible to monitor the snow status evaluating its dielectric constant by means of many techniques, which offer different advantages in terms of reliability, spatial and temporal resolution, impact on the current state of the snow and durability in time. The snow status can be also evaluated considering its conductivity. The conductivity can be related to many factors concerning the snow layer as temperature, stratification, crystal structure, density, liquid water content and conductivity of the relative fusion water. However in [22] it has been demonstrated that the snow conductivity is mainly a function of the snow density and of its air content.

2.4.1 TDR

The amount of liquid water content into the snowpack varies quickly during the time and it has also a sensible variation both in the horizontal and in the vertical direction. Therefore a method able to monitor continuously the snowpack with a large spatial resolution is fundamental in the understanding such variations. Among all the available techniques one of the most promising is the Time Domain Reflectometry (TDR) method. TDR estimates the dielectric permittivity by measuring the velocity of propagation of an electromagnetic wave, generated by a pulse generator, through a specific medium. The generated pulse propagates along a coaxial cable and enters the TDR probe that typically consists of a pair of metallic rods inserted into the considered medium. Part of the incident electromagnetic wave of the pulse is reflected at the top of the probe because of the impedance mismatch between the cable and the probe. The remainder of the wave propagates through the probe until reaching the end and then is reflected back to its source. The transit of the pulse for one round trip is typically measured with an oscilloscope. The return of the pulse is affected by the length of the probe or cable (travel distance) and by the permittivity of the insulator around the cable or probe (propagation velocity). If the physical length of the probe is known it is possible to determine the permittivity of the dielectric around the cable. Typically commercial instruments are in the range of 10 MHz to 1 GHz, with the central frequency around 200 MHz. The most common TDR sensors are quite small and allow therefore a non-destructive measurement of the permittivity of snow. However this translates into a reduced spatial resolution that makes necessary a large number of measurements and work to perform a complete evaluation of the snow status. Nevertheless, many authors have worked in the TDR field and have designed new sensors able to perform a complete measurement, continuous in time and with the necessary spatial resolution. One of these devices has been designed by [34] and consists of a thin-walled aluminum tube to be placed on top of a new

snow layer to be covered during the next snow fall. The sensor is light and white-colored in order to reduce the absorption of solar radiation. Moreover its design allows declining its height with respect to the ground according to the snow-pack settling. Such sensor showed good agreement with manual measurement performed in its surroundings. Another TDR sensor, with a different design, has been proposed both by [34] and [35], and consists of a flexible flat band cable up to about 100 meters in length, which can follow the settlement of the snow cover.

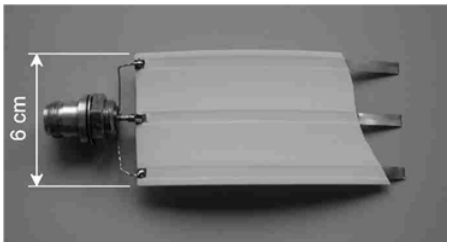

Figure 2.4: TDR flat band cable

Such long transmission lines can be permanently installed at the measurement site in different heights and enclosed by snow fall. The cables are insulated by means of white polyethylene (PE) in order to reduce the heating caused by the solar radiation.

Figure 2.5: Final configuration of the TDR measurement, with the cable deployed along the slope

However, air gaps, due to multiple freezing and thawing cycle, can develop around the sensor affecting the permittivity measurement. In order to prevent this effect and to correct the final result the cable is measured

twice, with small and large spacing leading to different measurement volumes. Thus the air gap has different effect on the volumes and it is possible to correct it, using a proper designed correction equation that relates air gap size and true snow permittivity. Such TDR sensor has been tested at low and high frequencies, leading to result in agreement with both manual reference measurement and lysimeter data taken at the reference site. Moreover such design allows to monitor a large volume and to monitor continuously the snow evolution during the seasons. However the TDR technique requires a separated measurement of the snow density for the determination of the dielectric snow characteristic of the snow.

2.4.2 Microwave sensors

The determination of the complex permittivity of snow allows the determination of the snow wetness and density. Therefore many instruments have been developed in order to be able to infer the dielectric properties of snow by direct measurement. The most important among all these instruments is the Snow Fork, developed in the Radio Laboratory of the Helsinki University of Technology, [Sih86]. The instrument consists of a resonator, realized by means of a parallel-wire transmission line resonator, open circuited at one end and short-circuited at the other end. The device can be pushed into snow or any other porous, granular or liquid material to be measured. The length of the resonator wires is a quarter of wavelength in the resonance. The resonance frequency has been chosen at 1 GHz in air, to obtain wires of reasonable length. The resonator is fed by high-frequency power through rigid coaxial cables and coupling loops. The cables are supported and coated by a glass fiber pipe that forms a solid stock. The coupling loops are protected by epoxy plastic. In order to make the device suitable to be pushed into the snow, even though a possible crust, the wires are made of stainless steel and are sharped at the end. Moreover the wires have been designed thin enough in order to perform the measurement in a non-destructive way, avoiding to

change considerably the snow density.

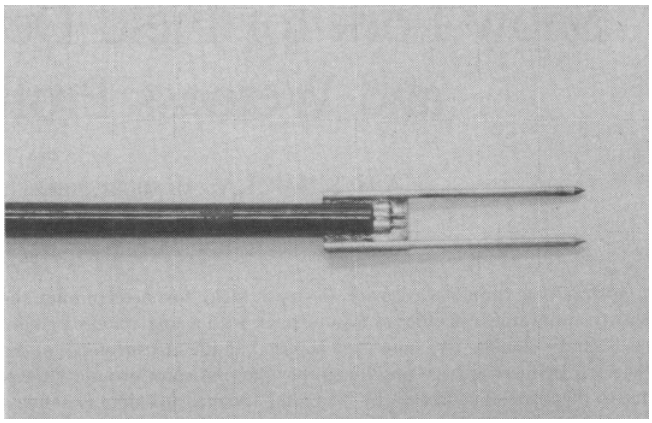

Figure 2.6: Snow fork in the design proposed by [42]

The complex permittivity is retrieved evaluating the change into the resonance curve of the resonator when it is pushed into the snow. In fact the real part of the snow permittivity lowers the resonant frequency whereas the imaginary part broadens the resonance curve, increasing also the attenuation at the resonance frequency (the quality factor is reduced). The range of measurement of the real part of the dielectric constant is [1, 2.9], whereas for the imaginary part the range is [0, 0.15]. The measurement system is automated and portable.

The results achieved by this instrument are precise and reliable, but in presence of very wet snow the measurement becomes quite complex to the wide broadening of the resonance curve. Moreover the results can be affected by errors due to the increase of density caused by the pressure of the device spikes in the material to be measured. The device also can give just limited information relative to the snow condition, due to its reduced spatial resolution. Moreover the device applicability is reduced to liquid water content between 0 and 10% with an error between 1-5%. Other instruments, [41], of this kind have been realized in the form of thin flat-plate sensors with differently sized and shaped coplanar conducting stripes. Also these sensors can be used for a non-destructive measurement of the snow permittivity. The

sensors work at 20 MHz, are battery-powered and can operate at ambient temperature down to $-10°C$. Different shapes of the sensors allow for measurement of the snow surface or of the snow volume. Also these instruments allow a precise estimation of the snow characteristics but they are affected by the same drawbacks of the Snow Fork and in particular they can guarantee a very limited spatial resolution.

2.4.3 Vertical Electrical Sounding

The method of vertical electrical sounding (VES) provides detailed information on the vertical succession of different conducting zones and their individual thickness and true resistivity. The measurements are made with a four-electrode array, consisting of two current and two potential electrodes (Schlumberger array). To reach investigation depth down to 100m, it is necessary to spread out the current electrodes at minimum up to 600m (AB-distance). This method was applied to measure the electrical conductivity of the snow during the winter of the 2005 and 2006 years, in several places of Pyrenees, Alps and Antarctica [22]. The 128 samples shown that the snow conductivity presents not much variability around $1\mu S/m$ (minimum 0.123, maximum 16.85, median 0.998). Moreover, data are independent of the height or orientation of the electrode array location and the electrode separation which determines the sensed snow depth.

2.5 GPS for channel characterization

The understanding of phenomenon related to the global climate change requires reliable information related to the Earth's cryosphere, which includes glaciers, sea ice, permafrost and frozen and snow-covered ground. The achievement of this kind of information has benefit in the recent year of the availability of a large satellite network provided by Global Positioning System (GPS). In fact, satellite remote sensing has proven to be particu-

larly effective for the continuous monitoring of large areas, that often would be also be inaccessible. The signal reflected from the ground and obstacles typically affect the quality of the signal received by GPS receivers. In fact the total signal arriving at the receiver would be the sum between the direct (Line of Sight- LOS) signal and the reflected one. The generated multipath contains information relative to the geometry of the reflector and to the dielectric constant of the reflector. Therefore the reflected signal can be used to infer snow characteristics and to retrieve snow depth.

2.5.1 GPS reflectometry for soil moisture retrieval

Earth sensing can be remotely performed taking advantage of the extraordinary GPS infrastructure built and maintained for navigation purposes. In fact the GPS signal reflected by the earth's surface can be collected and evaluated in order to infer information about the soil moisture and its dielectric constant. In particular the magnitude of the reflected signal is a function of the soil dielectric properties, with the possible additional interaction of other factors such as the surface roughness and the presence of vegetation that can introduce a certain level of attenuation. The relative permittivity in fact increases with the increase of the moisture content and it can be retrieved from the measurements of the soil reflectivity. The most suitable bandwidth to assess the soil properties lies in the L-band (1-2 GHz), as it guarantees low atmospheric attenuation, good penetration of the vegetation, independence from the solar illumination and a strong sensitivity to the soil moisture, [47], [48]. A possible configuration for the evaluation of the reflection generated by the GPS signal from the ground is the bistatic one, already used in radar applications. In such configuration the transmitter and the receiver are placed in different locations as reported in the following figure:

For a perfectly flat soil surface the expected reflection is specular, meaning that the incident and the reflection angles are equal in the plane of the transmitter and the receiver, the reflected power is coherent and regulated

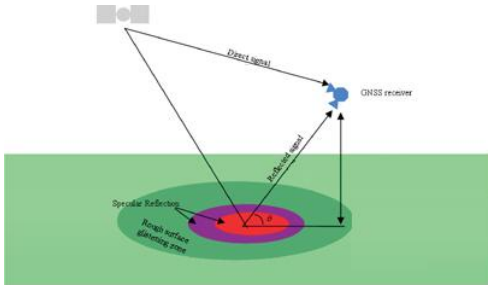

Figure 2.7: Bistatic GPS configuration

just by the Fresnel reflection coefficient of the active region from which the signal is reflected. The region involved in the reflection can be divided into Fresnel zones, with successive zones in phase opposition. As the elevation angle of the satellite decreases the Fresnel ellipses tend to become longer and narrower. On the other hand, if the surface is characterized by roughness, the coherent component of the reflected signal decreases and the surface generates also scattered incoherent power. As a result, for rough surfaces the active scattering region extends to a glistening region that surrounds the specular reflection region. The receiving antenna is typically able to measure both the LOS signal and the reflected one from the soil surface, making possible the timing and the possible calibration of the reflected signal with respect to the direct one. The received signal (P_{RX}) reflected by a generic surface is generally composed of a coherent component (PR_c) and a scattered incoherent component (PR_i):

$$P_{RX} = P_R^c + P_R^i \tag{2.2}$$

The coherent component of the reflected power can be written as:

$$P_R^c = \frac{P_{TX} G_{TX} G_{RX} \lambda^2 \Gamma}{4\pi (R_1 + R_2)^2 4\pi} \tag{2.3}$$

where P_{RX} is the transmitted power, G_{TX} and G_{RX} are the gain, respectively, of the transmitting and receiving antenna, $\lambda$is the wavelength,R_1 is the distance between the transmitter and the surface, R_2 is the distance

between the receiver and the surface and finally Γ is the power reflectivity. Such term decreases with the increase of the surface roughness and as the following expression:

$$\Gamma = |R(\theta)|^2 \chi(z) \tag{2.4}$$

where $\chi(z)$ is the characteristic function of the probability function of the surface heights z and $R(\theta)$ is the Fresnel reflection coefficient of the equivalent smooth surface. The reflectivity can be expressed as the combination of the vertical and horizontal polarization coefficients and it depends on the soil permittivity and, as a result, from the soil moisture content. Most of the measurements campaigns performed in the field of GPS soil reflectometry made use of airborne receiver and collected data for a period of time sufficient to test different soil conditions. The experiments conducted by [47], [48], demonstrated the existence of a spatial correlation between repeated tracks over the same areas and sensitivity to the field boundaries and differences in land cover type. Moreover the achieved results showed a good sensitivity to the soil moisture changes after precipitation events. Moreover [48] demonstrated that the GPS reflectometer data (direct signal, reflected signal and navigation data), properly processed, allow the extraction of the soil dielectric constant, starting from the Fresnel equation for normal reflectivity. The achieved results show good agreement between estimated and modeled values. Another study, [54], demonstrated the existence of a correlation between the Signal-to-Noise ratio at the receiver and the dielectric characteristic of soil, making the GNSS remote sensing technique a viable solution to infer soil moisture content.

2.5.2 GPS reflectometry for snow characterization

The received GPS signal has been recently exploited by many authors to develop algorithm useful for the retrieval of the snow characteristics. In [60] has been developed a model in which the relative power received by the

GPS antenna is used as the fitting-function in a Quasi-Newton Algorithm (QNA). Such algorithm is used to estimate, in a least-square sense, to non-linear parameters: the snow depth and the snow density. The model for calculating the received power is based on the use of a vertically mounted hemispherical directional antenna with no side lobes, a smooth snow layer of infinite extent, a ground reflector of infinite extent (in the experimental proofs a small ground reflector has been used) and uniform plane waves with a monochromatic frequency. The considered antenna has the maximum directed toward the horizon (zenith), in order to have equal gain from the direct and the reflected signal. However, in such configuration the received GPS signal increases with decreasing elevation angle, as the antenna gain pattern increases with decreasing elevation angle. Moreover at low elevation angles the effect of the reflected signals is maximized because the electrical path of the GPS signal in the snow increases as the elevation angle decreases. The selected antenna has been designed to suppress multipath, even if it does not completely remove it. The received signal is modeled as:

$$P = |1 + \frac{r_h + r_v}{2} exp(i\phi)|^2 \tag{2.5}$$

Where $r_h$and r_v are the field reflection at the horizontal and vertical polarization, respectively, $\phi = \frac{4\pi h \sin\theta}{\lambda_0}$ is the phase shift difference in physical path length between the direct and the reflected path, h is the height of the antenna, θ is the elevation angle and λ_0 is the GPS free space wavelength.

The geometry of the system is reported in the following figure:

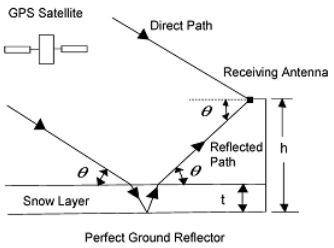

Figure 2.8: Geometry of the system for the retrieval of the snow characteristics by GPS reflectometry, [60]

The model allows the computation of r_h and r_v by means of a ray diagram. Such quantities are related to the complex permittivity and to the thickness of the considered dielectric. The evaluation of the snow depth and density is performed by means of an iterative procedure until reaching the best estimation; several guess are required to determine the best least-square estimate. The obtained results are in good agreement with the theory. However the model should be improved taking into account additional information, such as the antenna pattern, the reflection from the surrounding and eliminating the simplified model of perfect flat air-snow interface.

Another model developed by [61] exploits the multipath effect on the GPS Signal-to-Noise Ratio (SNR) in order to determine snow parameters. According to the proposed model the SNR data obtained with a horizontal reflector is equal to:

$$SNR = A\cos(f\sin E + \phi) \tag{2.6}$$

where E is the satellite elevation angle, A is the amplitude and depends on the reflector's dielectric constant, on the surface roughness and on the gain pattern of the antenna. The frequency f depends instead on the transmitted GPS frequency, on the height of the antenna, on the snow density and on the moisture of the underlying soil. The application of well-known theoretical models allows the matching between the frequency and the snow depth, taking into account the different conditions of snow and soil. The comparison between the GPS SNR observations and the model prediction for f shows good agreement: the multipath has a sensible longer period in presence of snow with respect to the bare soil. The performed measurement shows a sensitivity of the SNR to the different snow depth, even if it is not possible to directly measure it. To convert the GPS data into snow depth it is necessary to use the Lomb-Scargle periodogram to estimate the multipath peak frequency and then apply some theoretical model to match the relationship between snow depth and frequency for various snow densities. In conclusion

the obtained results agree well with temporal measurement of snow depth performed by means of ultrasonic sensor. However the proposed model introduced some simplifications that limit its application to some particular snow and environment conditions. In fact the model assumes planar layer of snow (not realistic) and makes use of low elevation angles for the estimation, which maybe are not the best to snow sensing activity. Moreover note that the models require some kind of estimation of both snow permittivity and density.

In conclusion, the development of satellite remote sensing technique is very promising and the results achieved are encouraging (errors of ± 5 cm), [61], [59], however the reviewed techniques are not actually providing data with accuracy comparable of that of other remote sensing techniques. Nevertheless methods based on the use of GPS are cheap and simple and the future development of new additional frequencies makes them very attracting for the design of new technique for the cryosphere monitoring.

2.5.3 GPS propagation in snow

GPS signal is able to penetrate and propagate into the snowpack very well. Therefore this important aspect can be exploited to infer snow parameters or even to develop tool for the identification of victims buried by avalanches. However the propagation into the snow is a complex phenomenon that has to be deeply analyzed. First of all, most of the attenuation suffered by the GPS signal is due to the reflection at the interface between the air and the snow cover. The penetration of the signal can be improved with smaller incidence angle, and thus the highest reflection losses are associated to low elevation satellites. However reflection at the surface interface is not the only cause of signal degradation, as further attenuation within the snowpack is introduced by dielectric losses, fading and scattering. In particular, the amount of liquid water content plays a crucial role in the propagation of the signal through the snow. In fact, particles of water and ice reflect the signal

and generate scattering, which sensibly degrades the signal quality. The ability of a signal to propagate into the snowpack is directly related to its water content: dry snow has in fact a penetration depth of around 400 meters at 1.5 GHz, while wet snow has only a penetration depth of 3 meters at the same frequency, [68]. A preliminary study performed by [64] has the aim of understanding the feasibility of a transmission system based on GPS for the rescue of avalanche victims. The experiment used standard sensitivity GPS receivers, placed under incremental layers of compacted snow. The snow was intentionally compacted to simulate the avalanche conditions. The measured parameters include the signal quality, the number of tracked satellites and the received signal strength. The achieved results show that the GPS signal could be received up to a depth of 1 meter into compact packed snow using the available commercial receivers of that period. The mentioned study was the first approach to this topic and demonstrated the viability of such techniques. In the recent years a huge technological improvement in the field of the GPS receivers makes available on the market the so-called High Sensitivity GPS receivers. Such devices are able to track the satellites signals 20-25 dB below the threshold of conventional receivers, making possible the acquisition of the signal also in harsh environments or in presence of obstructions. Such devices always guarantee positioning availability, but the achieved precision is of course lower than that obtained by LOS observations. However the new features of the high sensitivity receivers make them attractive and therefore many studies, [66], [67], [69], have been dedicated to the evaluation of the performance of this kind of receivers into the snowpack. The study conducted by [67] demonstrated the capability of the High Sensitivity GPS receivers of tracking a sensibly higher number of satellites with respect to traditional GPS receivers. However in the mentioned study has been also underlined the fact that tracking satellites with a poor SNR can induce significant errors in the computation of the position. For what concerns the evaluation propagation into the snow pack, the GPS signal, as reported in Figure 2.9, can take many

different paths within the snow pack, depending on the effect of multipath and fading due to the density changes and the difference between layers.

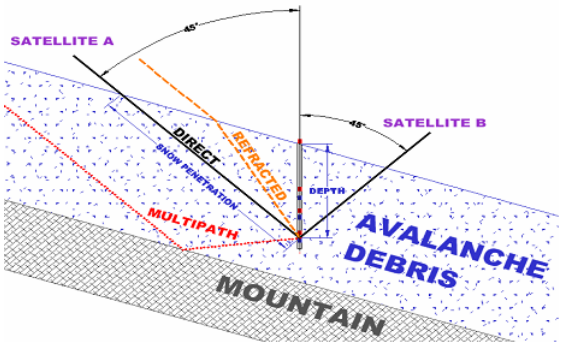

Figure 2.9: The GPS signal can take different paths into the snowpack depending on the different layers and incident angles, [Sch06]

The testing activity of [66], [69] was based on the evaluation of different high sensitivity receivers buried at different depths into the snow pack. The monitored parameters include the pseudorange measurement, the number of tracked satellites, the signal to noise ratio and the achieved position accuracy. The results show that the high sensitivity GPS receivers are able to track the signal through 15 meters of avalanche deposited snow, with a mean attenuation of 1.8 dB for each meter of snow. The resulting position estimation is not enough precise to allow the detection of a buried victim for rescue purposes, even if applying a simple averaging algorithm improves sensibly the system performance. The mentioned system also tested a Zigbee device at 2.4 GHz in order to communicate with the surface; however such link proved to be unreliable beyond 10 meters.

Chapter 3

Snow water content evaluation for avalanche forecasting

3.1 Avalanches

Avalanches represent one of the major hazards related to mountain activities because of their unpredictability and destructive power. Wordwide, every year, 150 people are killed by avalanches, mostrly among skiers and snowboarders. Statistics demonstrate that the 93% of victims survive if rescued in the first 15 minutes, whereas after this small interval of time the rate of survival drop fast. After 46 minutes only 20-30% of victims are alive. An avalanche is a complex phenomenon determined by many factors, including the slope characteristics (inclination, roughness, etc), the weather conditions and the corresponding snowpack variations. The different combinations of these factors create low, moderate, considerable, and high avalanche hazards. Avalanches can be divided into three main categories, depending on the type of snow involved. The powder avalanche is generated in a single point and the volume of powder snow involved increases as it progresses. Slab avalanches are caused by wind, which creates on lee slope unstable layers of snow, or by the presence of loose snow layers with a smooth or damp surface. The slab avalanche starts as a consequence of a fracture, generated even by the weight of a skier, making the top layers sliding over the lower ones. Finally,

wet snow avalanches, instead, are typically determined by snow exposure to high temperature, which reduces the cohesion properties of snow, which melt and start percolating. Moreover they can be triggered by rain and occasionaly they can occur due to glide on, for example on an impermeable rock bed. Such avalanches start from a single point, spread as they slide and, even if generally slow, they can have a high destructive power due to snow density. It has been demonstrated [Schneebeli (2004)] that the release of a wet snow avalanche is related to the water flow in the snowpack. In fact, a snowpack is typically composed by different and heterogeneous layers as a consequence of metamorphism due to successive snowfalls, melting and freezing Such layers have an important influence on the movement of water flow through the snowpack. In general, boundaries between two snow layers of different textures can either impede or accelerate downward flow. The vertical water movement through the snow cover might as much be affected by the existence of capillary barriers due to fine-overcoarse layering as by impermeable ice lenses or crusts. Below a volumetric water content of about 7% the mechanical strength of the snow pack seems not to decrease significantly, whereas increased wet-snow avalanche activity has been observed if the volumetric water content exceeded 7%. Therefore the evaluation of the liquid water content into the snowpack constitutes one of the variables to take into account in the identification of possible avalanche hazard. The capability of constantly monitoring the snow status and in particular its liquid water content can thus represent an useful and efficient system to forecast avalanches

3.2 Snow characteristics

Dry snow can be considered, from the electromagnetic point of view, as a heterogeneous medium composed of ice and air. The dielectric constant of ice do not vary in a wide range of frequencies (between 10 MHz and 1000 GHz), and thus the permittivity of dry snow is dominated by the snow density effect.

Many models have been developed in order to determine an exact expression to relate permittivity and density. The resulting expressions give a linear dependency between the real part of the dielectric constant of dry snow and its density. The imaginary part, instead, is quite small, [36]. On the other hand, wet snow is a dielectric mixture of ice, liquid water and air whose behavior depends both on frequency and density. Moreover wet snow is characterized by two different regimes of liquid water saturation: the pendular regime and the funicular regime. The pendular regime is characterized by the presence of a continuous distribution of air in the pore structure and by the presence of isolated inclusions of liquid water. Such regime correspond to a low level of liquid saturation, in general less than 7%. The funicular regime corresponds, instead, to a higher level of liquid saturation in which liquid water is continuously distributed in the whole pore space and the air is trapped into isolated bubbles. The dielectric constant of wet snow in the pendular regime is independent on the snow structure and the behavior seems to be dominated by the water effect. The transition between the pendular and the funicular region represents a critical point for the evaluation of the snow wetness and for the validation of the proposed empirical models. The relation between the dielectric constant and the snow wetness is a transcendental function that can be expressed in terms of Taylor series expansion. The approximation to the first order term can be considered sufficient for the purpose of this work, even if the introduction of a quadratic dependence allows to achieve more consistent results. In particular the real and the imaginary part of the dielectric constant have the following expressions:

$$\epsilon'(f\, m_v) = \epsilon'(f\, m_v = 0) + \alpha'_1(f) m_v + \alpha'_2(f) m_v^2 \tag{3.1}$$

$$\epsilon''(f\, m_v) = \sigma(f\, m_v = 0) + \alpha''_1(f) m_v + \alpha''_2(f) m_v^2 \tag{3.2}$$

where m_V is the liquid water content, expressed in percent per volume, and the α terms assumes, respect to the frequency of interest, the following

values:

	f = 900 MHz	f = 2.45 GHz
α_1'	0.092	0.1
α_2'	0.83	0.005
α_1''	0.012	0.01
α_2''	0.104	0.0005

Table 3.1: α parameters at 900 MHz and 2.45 GHz

The corresponding values of the dielectric constant, calculated for different level of snow wetness, are reported in the following tables:

Medium	Water Content (Vol.%)	ϵ'	σ
ice	–	3.15	0.0001
dry snow	0	1.6	0.0001
moist snow	0-3	1.85	0.0062
wet snow	3-8	2.2	0.016
very wet snow	8-15	2.7	0.031
soaked snow	>15	3.6	0.045
water	–	81	0.49

Table 3.2: Dielectric characteristics of different type of snow at 900 MHz

Medium	Water Content (Vol.%)	ϵ'	σ
ice	–	3.15	0.0001
dry snow	0	1.47	0.0001
moist snow	0-3	1.71	0.0068
wet snow	3-8	2.1	0.0299
very wet snow	8-15	2.9	0.0708
soaked snow	>15	3.3	0.0953
water	–	81	2.8583

Table 3.3: Dielectric characteristics of different type of snow at 2.45 GHz

3.3 The proposed system

The developed activity has been devoted to the realization of a sensor able to permanently and continuously monitoring the snow pack in order to deduce information about its liquid water content. Therefore the proposed model is based on the exploitation of the electromagnetic signal generated by properly designed antennas, [74] ,[75]. In particular the model of the sensor has been configured as an alignment of electric and magnetic radiators, as reported in 3.1 and 3.2.

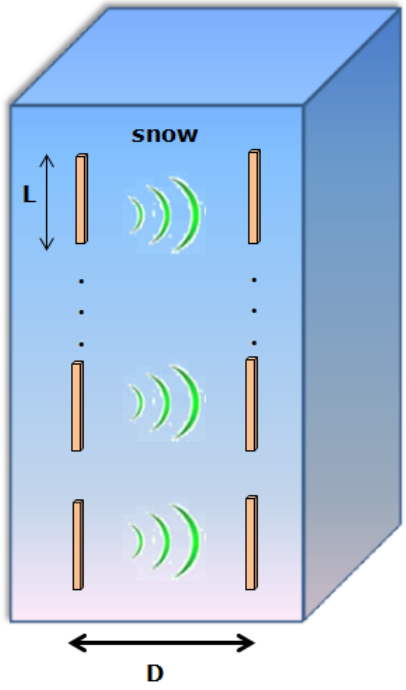

Figure 3.1: Schematic view of the sensors buried into the snow - electric radiators

Each radiator is fed individually, as the whole alignment is not intended as an array. By monitoring the relative attenuation among the two separate alignments of vertical antenna it is possible to deduce information about the relative permittivity and electrical conductivity with sufficient precision. Moreover the variation into the strength of the received signal is strictly related to the content of liquid water present into the snow pack. The real-

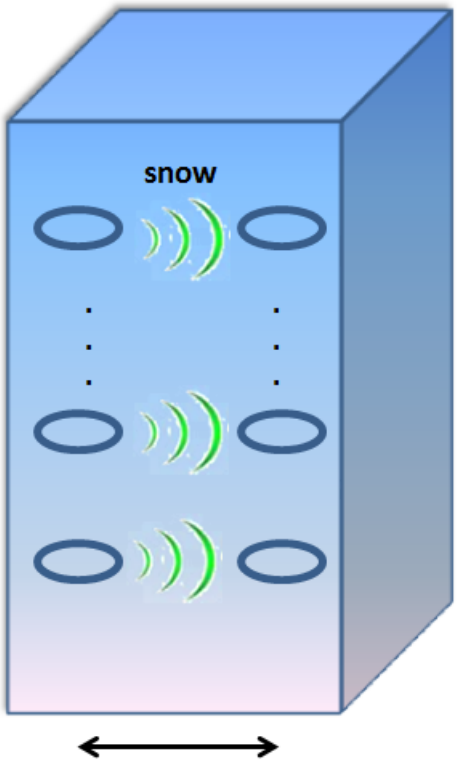

Figure 3.2: Schematic view of the sensors buried into the snow - magnetic radiators

ization of two alignments of radiators guarantees the continuous monitoring of the snow wetness. Moreover such system has also the advantage of allowing the evaluation of the dielectric parameters at different depths, taking into account the possible presence of layers with different characteristics. The system has been designed in order to reduce at minimum the variation in density introduced to the snow by the insertion of the sensor itself.

3.4 The simulations

The proposed model has been tested and validated by means of CAD simulations performed with HFSS, a software for the electromagnetic modeling by Ansoft. HFSS is a simulation tool for 3-D full-wave electromagnetic field simulation based on either the proven finite element method or the well established integral equation method. The simulations had the objective of

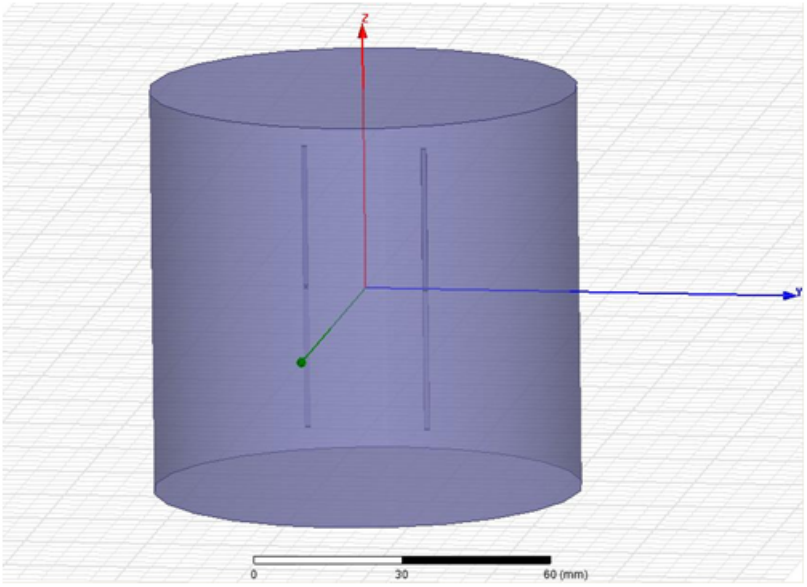

Figure 3.3: Example of the simulation configuration with dipoles in HFSS

testing the proposed sensor model with different settings in order to identify the best configuration. The system has been tested using both dipoles and loops. Moreover the system has been tested considering the variation of the following parameters:

- the length L of the dipole $[0.1\lambda, \lambda]$
- the circumference C of the loop $[0.3\lambda, \lambda]$
- the distance D between the radiators $[\lambda/5, 3\lambda]$
- the frequency f (900 MHz and 2.45 GHz)

Moreover the system has been changed varying the boundaries conditions in order to simulate the presence of different type of snow, characterized by different wetness levels.

3.5 Results

The obtained results show a dependence of the received power from the snow water content. In fact, with both radiators, the availability of at least 10 dB of dynamics, in terms of received power, makes possible the identification of the snow characteristics using such parameter. The main results are reported in the following tables, where D is the distance between the radiators, L is the dipole length and C is the circumference of the loop. The values of received power are expressed in dB and they are all normalized with respect to the power received considering as medium the free space.

		f=900MHz $D = \lambda/5$ $L = \lambda/2$	f=2.45GHz $D = \lambda/5$ $L = 0.7\lambda$	f=900MHz $D = \lambda/4$ $L = \lambda$	f=2.45GHz $D = \lambda$ $L = \lambda/2$
Medium	Water Content (%)	$P_{RX}[dB]$	$P_{RX}[dB]$	$P_{RX}[dB]$	$P_{RX}[dB]$
dry snow	0	-0.2653	-4.7569	-6.3034	-2.3287
moist snow	0-3	-0.4689	-7.9195	-8.4833	-3.7902
wet snow	3-8	-2.7080	-12.8903	-13.8059	-7.3849
very wet snow	8-15	-6.0703	-20.8426	-16.0781	-12.9715
soaked snow	> 15	-10.8786	-21.8537	-16.1558	-13.1729
water	-	-47.4341	-38.7873	-46.6324	-44.6983

Table 3.4: Normalized power received by the system with different configurations of dipoles at 900 MHz and 2.45 GHz

		f=900MHz $D = \lambda$ $C = 0.4\lambda$	f=2.45GHz $D = \lambda/2$ $C = 0.4\lambda$	f=900MHz $D = 0.7\lambda$ $C = 0.3\lambda$	f=2.45GHz $D = \lambda/3$ $C = 0.7\lambda$
Medium	Water Content (%)	$P_{RX}[dB]$	$P_{RX}[dB]$	$P_{RX}[dB]$	$P_{RX}[dB]$
dry snow	0	-2.9853	-4.3799	-2.8796	-0.5031
moist snow	0-3	-3.1551	-4.6618	-4.1294	-2.8824
wet snow	3-8	-7.5726	-8.5146	-5.8226	-4.8648
very wet snow	8-15	-13.5313	-13.0536	-8.6001	-10.0841
soaked snow	> 15	-13.7456	-13.5658	-31.2033	-11.6710
water	-	-53.7150	-27.4256	-36.3994	-22.1184

Table 3.5: Normalized power received by the system with different configurations of loops at 900 MHz and 2.45 GHz

3.6 Conclusion

The proposed system has been tested by means of HFSS simulations in many different snow conditions and considering a wide range of possible configurations. The system is able to guarantee at least 10 dB of dynamics between dry and soaked snow, which represents a sufficient range to be able to detect different snow conditions. However the realization of a prototype to be testes in real snow conditions is necessary to validate the proposed scheme.

Chapter 4

Snow cover characterization by GPS multipath signals

4.1 The project

The methods available in literature for the assessment of the snow characteristics are able to give an estimation of the different parameters with different level of accuracy and different resolution in time and space. However all the techniques are affected by drawbacks that can be divided into two main categories:

- high cost: many described techniques achieve excellent results in terms of accuracy but requires the setup of a high cost infrastructure or are airborne techniques;
- reduced spatial and/or temporal resolution: other techniques cannot guarantee the snow monitoring status to a large scale or for long period of time.

The exploitation of the GPS signals has been considered as a possible answer to address these issues. In fact making use of the expensive infrastructure built-up for navigation purposes and exploiting the GPS signal characteristics could conjugate both low cost and long term monitoring of a large portion of space.

The idea developed by the GTE (Grupo de Tecnologias en Entornos Hostiles) of Universidad de Zaragoza was to use commercially available GPS device to monitor the snow cover status. The devices have to be deployed both on the snow surface and under the snow cover in order to deduce the snow depth and the SWE, as reported in 4.1.

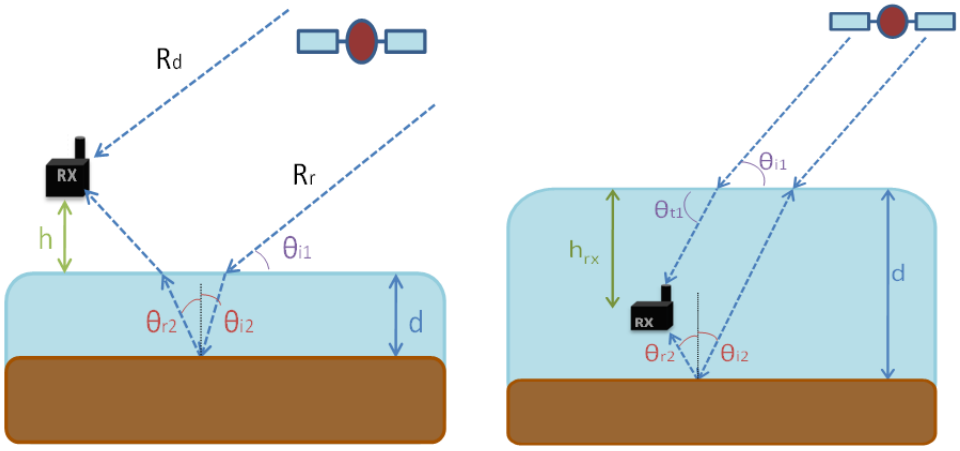

Figure 4.1: Model of the proposed system

However, in order to verify the feasibility of the system, a bunch of experiments has been performed in a controlled environment. The first experiment has been performed on the roof top of Universad de Zaragoza, with the GPS receivers in standard conditions. The following experiments have been performed in different conditions, with the GPS surrounded by different dielectrics, in order to verify the sensibility of the GPS signal to the variation of dielectric medium. In the following sections each experiment will be detailed and for each of them the experimental results will be displayed with graphs.

4.2 Experimental activity

The experimental activity has been performed on the roof top of Univerdad de Zaragoza, in order to guarantee to the GPS receivers the best visibility conditions. The first bunch of receivers were equipped with three Sanav GPS patch antennas, which characteristics are reported in 4.2. For the

last experiment two additional Beyondoor GPS-GLONASS patch antennas were available. Their characteristics are reported in 4.3.

Antenna features	
Center Frequency	1575.42 MHz +/- 1.023 MHz
Polarization	R.H.C.P. (Right Hand Circular Polarization)
Absolute Gain at Zenith	+5 dBi typically
Gain at 10 degs Elevation	-1 dBi typically
Axial Ratio	3 dB max
Output VSWR	1.5 max
Output Impedance	50 ohm

Figure 4.2: Sanav patch antenna features

Antenna features	
Frequency	1575-1602 MHz
Center Frequency	1590 MHz
Polarization	R.H.C.P. (Right Hand Circular Polarization)
Peak Gain	> 3dBi
Gain Coverage	> -4 dBi at $-90 < \theta < 90$
Bandwidth	±5 MHz
VSWR	1.5:1
Output Impedance	50 ohm

Figure 4.3: Beyondoor patch antenna features

The GPS antennas were connected to the corresponding GPS acquisition cards. The receivers system was connected to a rugged pc by means of an active hub and the GPS data were recorded by means of the Fastrax software, provided by the receivers' manufacturer. The data were collected in form of

three NMEA sentences:

- GPGGA
- GPGSA
- GPGSV

The content of the three GPS frames is reported in 4.4.

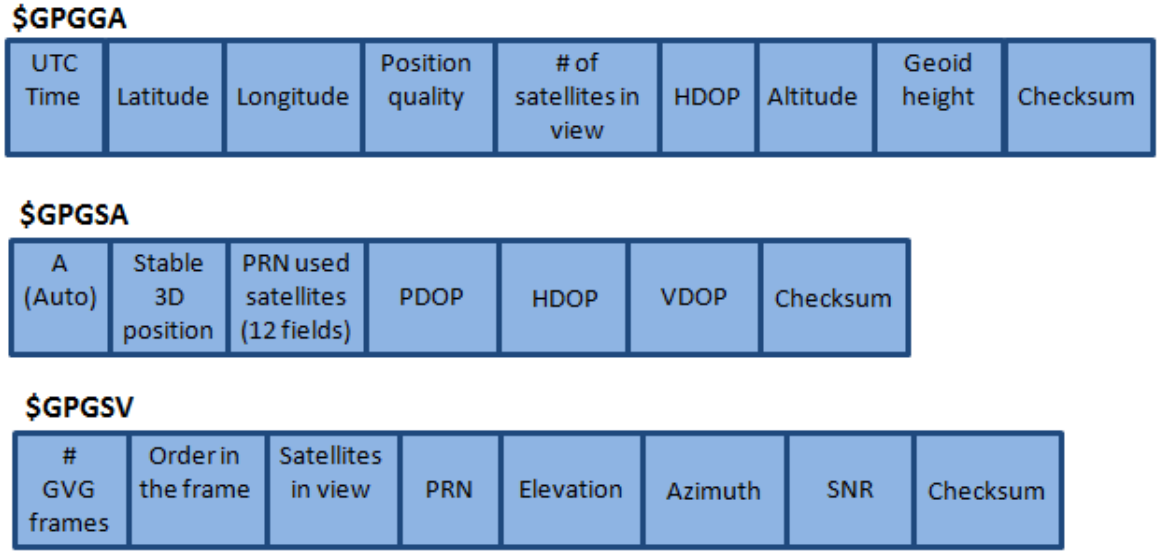

Figure 4.4: Content of NMEA navigation messages

The data collected are all that related to the position (latitude, longitude and elevation), the PRN of the satellites in view, the corresponding acquired SNR (Signal-to-Noise-Ratio) and the quality of the signal data (HDOP, VDOP, PDOP). The analysis of all this information allowed to evaluate the impact of different surrounding dielectric conditions on the GPS signal and, as a consequence, the feasibility of the proposed approach. The analysis of the huge amount of data produced during the different acquisitions has been performed by means of a Matlab program, reported in appendix B.

4.2.1 Free space

The first experiment had the objective of verifying the coherence of the data collected by the three different GPS receivers, both in terms of recorded position and signal strength. Therefore the receivers have been placed on the roof of one of the building of Universidad de Zaragoza to collect data for 24 hours. The antennas were placed on top of a metallic plate in order to reproduce their ideal working condition and connected to a rugged pc by means of an active hub. In 4.5 is reported the experimental setup.

Figure 4.5: Scheme of the free space experiment set-up

In the following graphs are reported the recorded data in terms of position (latitude, longitude and altitude) and the signal strength with respect to the time. The data are referred to an interval of 7 hours, as a sample of the 24 hours acquisition. The SNR is reported, as an example, for the satellite with PRN 14, among all the available satellites in view. The experiment shows that the collected data are coherent of the three receivers and therefore it has been possible to proceed with the other experiments. Moreover the data relative to the position quality confirmed the good configuration of the system for the position computation.

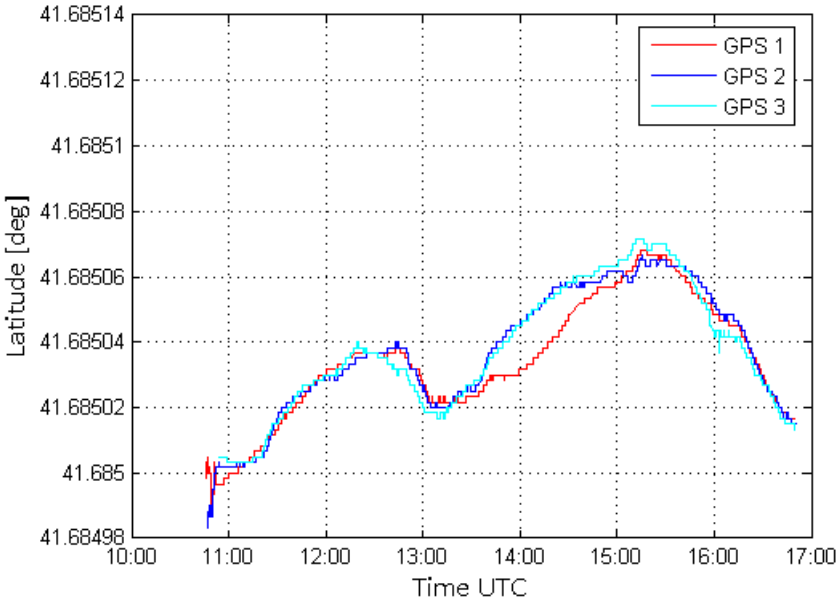

Figure 4.6: Computed latitude [degs] for the three GPS receivers with respect to the time [UTC]

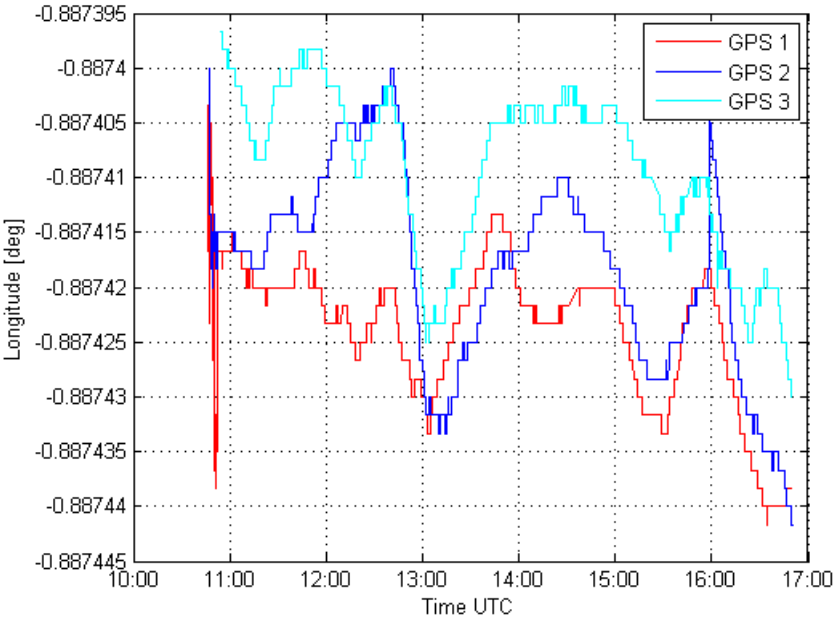

Figure 4.7: Computed longitude [degs] for the three GPS receivers with respect to the time [UTC]

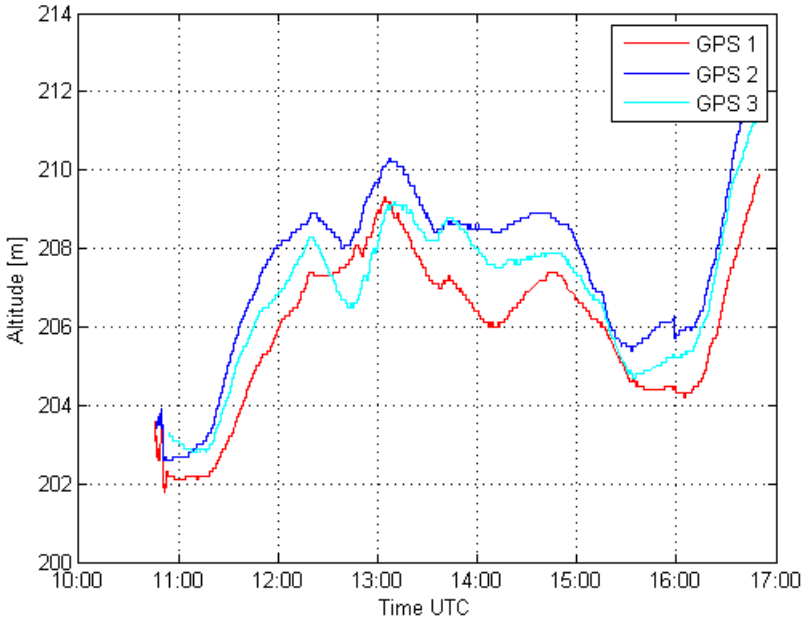

Figure 4.8: Computed altitude [meters] for the three GPS receivers with respect to the time [UTC]

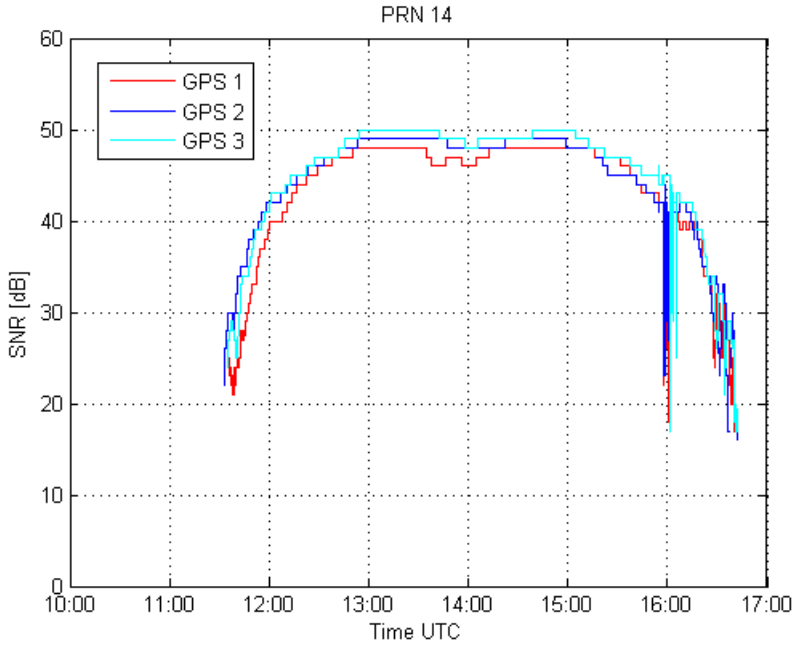

Figure 4.9: Computed SNR [dB] for the satellite PRN 14 for the three GPS receivers with respect to the time [UTC]

4.2.2 Water

The second experiment has been performed in order to evaluate the impact of a diectric on the reception of the GPS signal. The chosen dielectric has been water, in order to test the worst case of the possible scenarios of the considered application. The evaluated parameters were:

- the recorded position in terms of latitude, longitude and altitude
- the Signal-To-Noise Ratio(SNR) recorded by three GPS receivers for the available in view satellites
- the resulting dilution of precision in the horizontal and vertical direction (HDOP and VDOP)

The experiment has been performed placing the GPS receivers (GPS IT500) on the roof-top of the I+D building of the Universidad de Zaragoza. The data recording lasted for 24 hours, from 11 a.m. of July, the 16th 2012 to 11 a.m. of the following day. The three receivers were placed according to the configuration reported in the following figures.

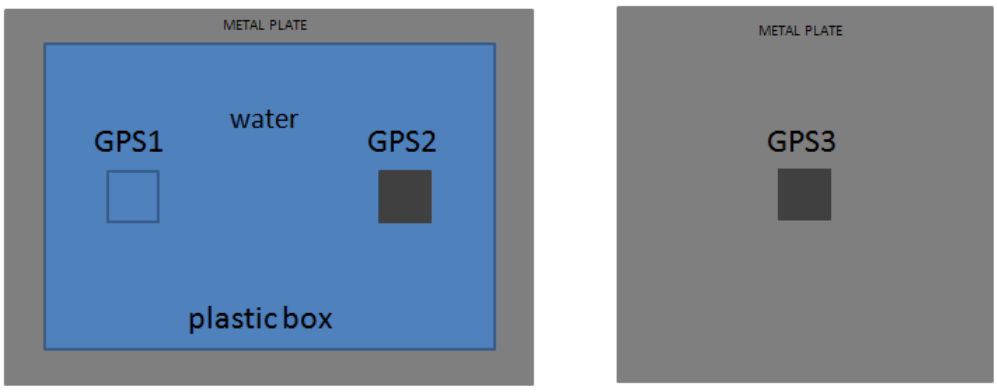

Figure 4.10: Upper view of the used measurement configuration

All the receivers have been placed over a metallic plate. GPS1 has been placed under a plastic container filled with the 2 centimeters of water, whereas GPS2 has been placed inside the container. On the other hand,

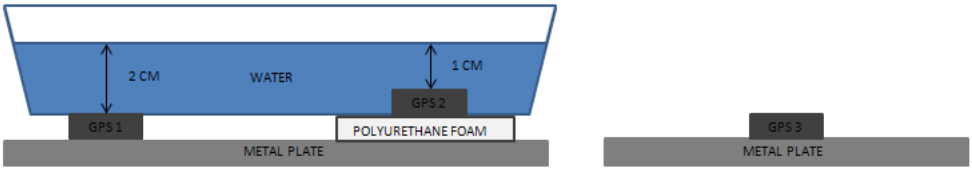

Figure 4.11: Lateral view of the used measurement configuration

GPS3 has been placed over a separated metal plate, in a standard configuration, to be used as a reference. All the receivers have been connected to a Rugged PC by means of the corresponding acquisition boards, using an USB hub. The NMEA sentences have been recorded using the acquisition software Fastrax GPS Workbench 5, given by the modules manufacturer.

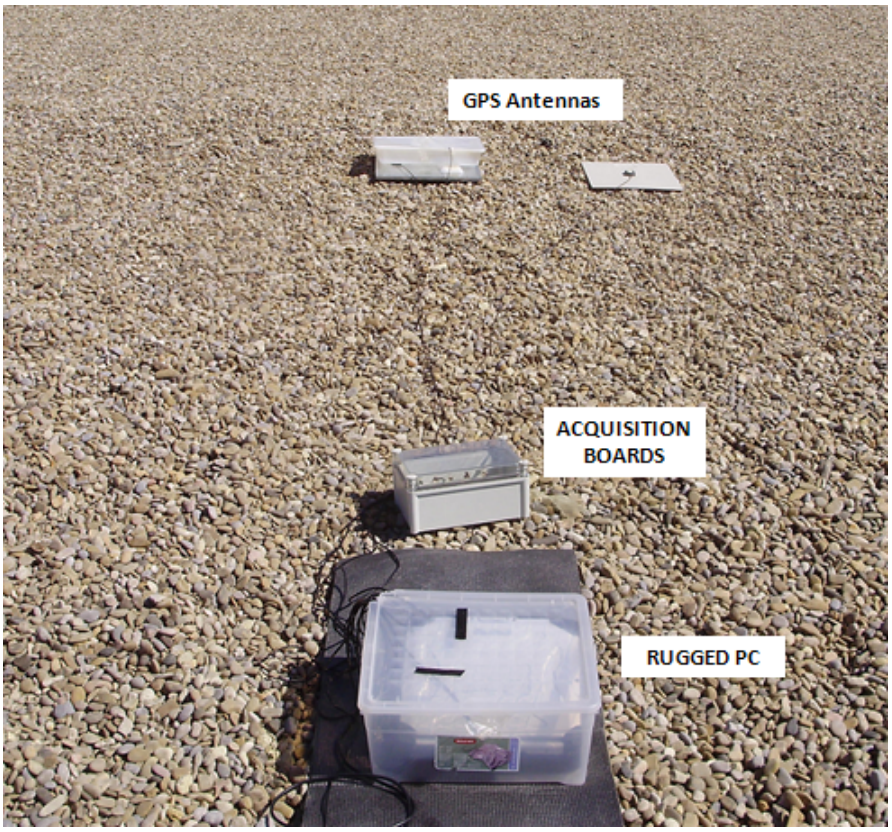

Figure 4.12: Front view on the experimental setup

Figure 4.13: Detail of the GPS receivers configuration. On the right GPS1 and GPS2 inside and under a box containing water and, on the left, GPS3 in a standard free space configuration

Figure 4.14: Detail of the box filled with water

As can be observed in 4.14, the container was covered with a plastic layer to avoid water evaporation as a consequence of high temperatures (22°C-38°C).

In the following are reported the graphs obtained processing the GPS output data with Matlab. The graphs are referred to the 12 hours acquisition performed during night-time that results to be less affected by noise and acquisition errors. In fact the data obtained during the day seem to be affected by interference, whereas the night-time results present just some spikes due to acquisition errors, which have been eliminated with a filtering function in Matlab. In the following graphs the filtered data have been underlined with dots.

The following graphs report the latitude, longitude, altitude, position (East and North) computed by the GPS receivers.

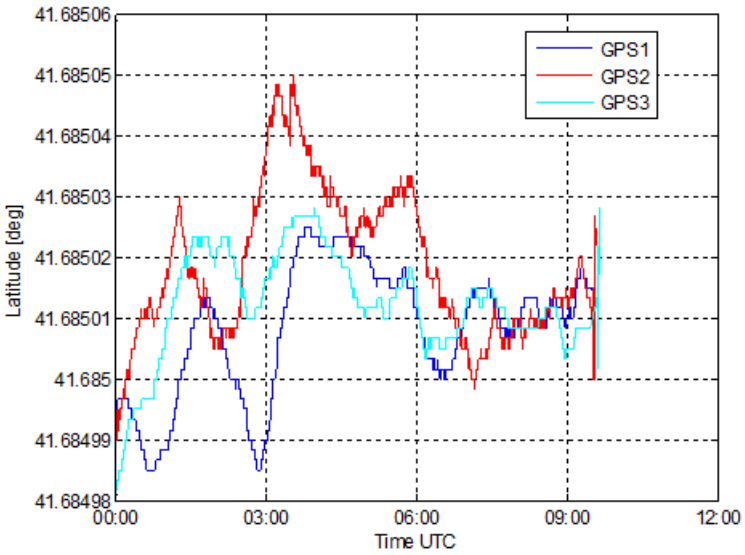

Figure 4.15: Comparison between the latitude (deg) with respect to time (UTC) recorded by the 3 GPS modules

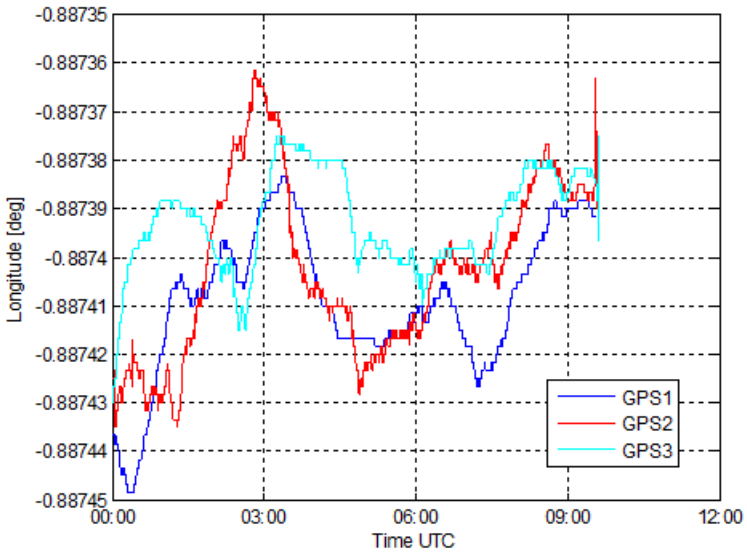

Figure 4.16: Comparison between the longitude (deg) with respect to time (UTC) recorded by the 3 GPS modules

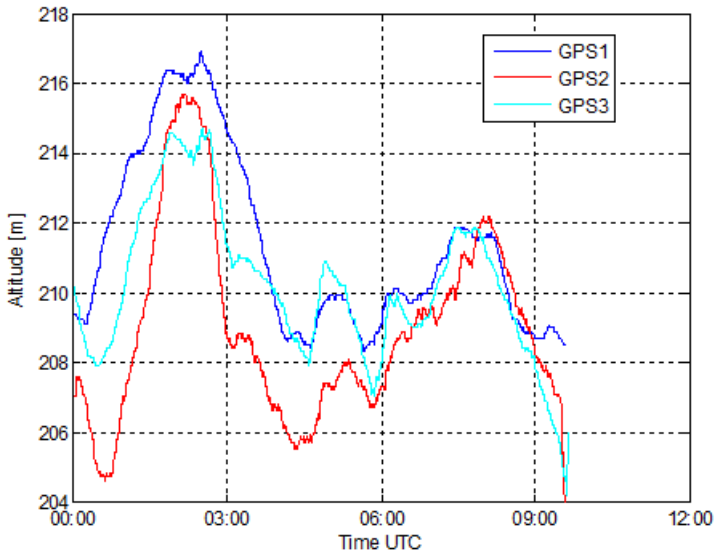

Figure 4.17: Comparison between the altitude (meters) with respect to time (UTC) recorded by the 3 GPS modules

The evaluation of the computed positions shows that the best agreement is obtained between GPS1 and GPS3. In fact, GPS2 antenna is completely immersed into the water and it is not in direct contact with the metal plate (causing probably some antenna mismatch to be further analyzed, as the antenna is designed to be placed on cars rooftop).
The following graphs represent the SNR variation of the different in view satellites recorded by each GPS module as a function of time. The reported graphs include the satellites in view for the longest period and with the best performance in terms of SNR (PRN 2,9,15 and 17).

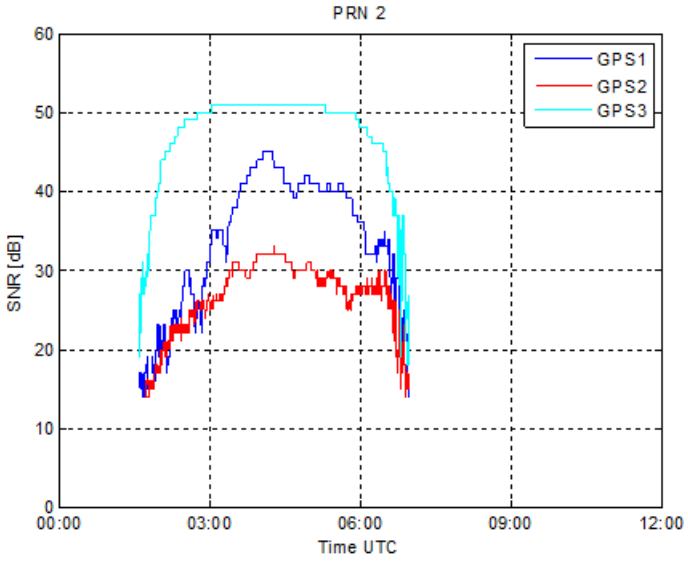

Figure 4.18: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 2

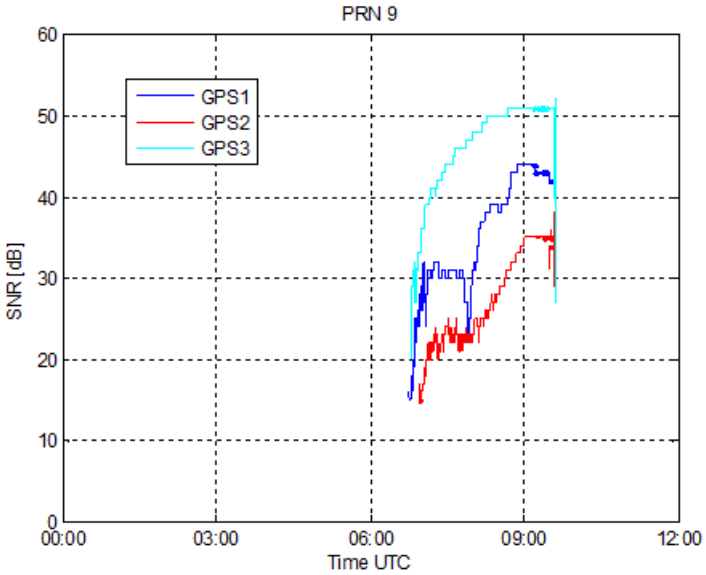

Figure 4.19: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 9

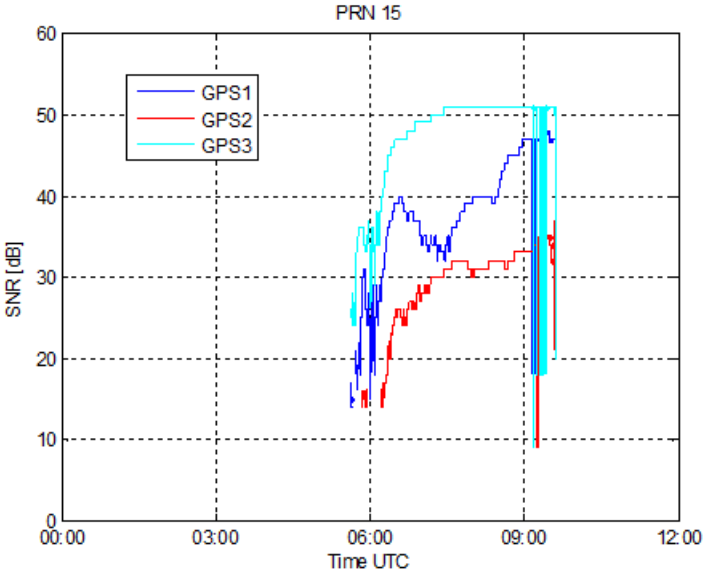

Figure 4.20: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 15

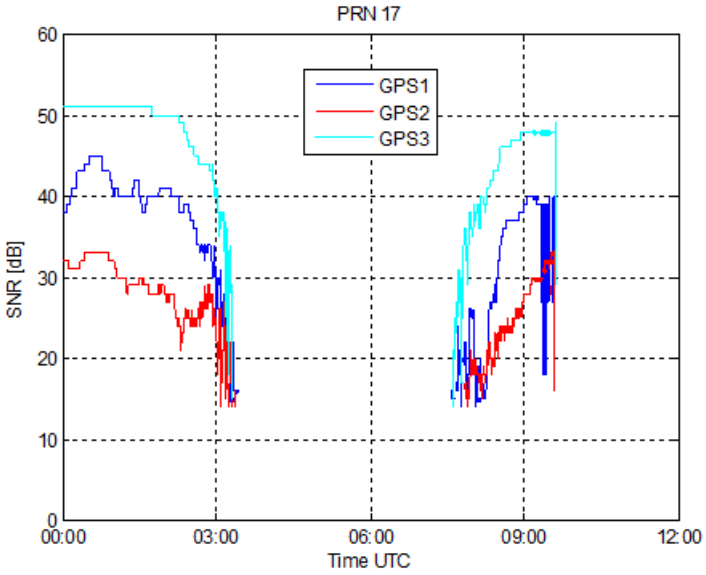

Figure 4.21: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 17

The reported graphs for the SNR obtained for each GPS modules show clearly the effect of water: the resulting loss for all the satellites in view is around 10 dB for GPS1 (under the container) and around 20 dB for GPS2 (into the container). Again the GPS2 is strongly affected by the presence of the water, reaching a difference in SNR with respect to GPS1 on average around 10 dB, with a peak of 13 dB, as reported in Figure 11 (4.00 a.m. UTC time).

The following graphs report the SNR variation recorded by the GPS modules for satellites with PRN 2,9,15 and 17 with respect to the elevation (degrees) and the azimuth (degrees).

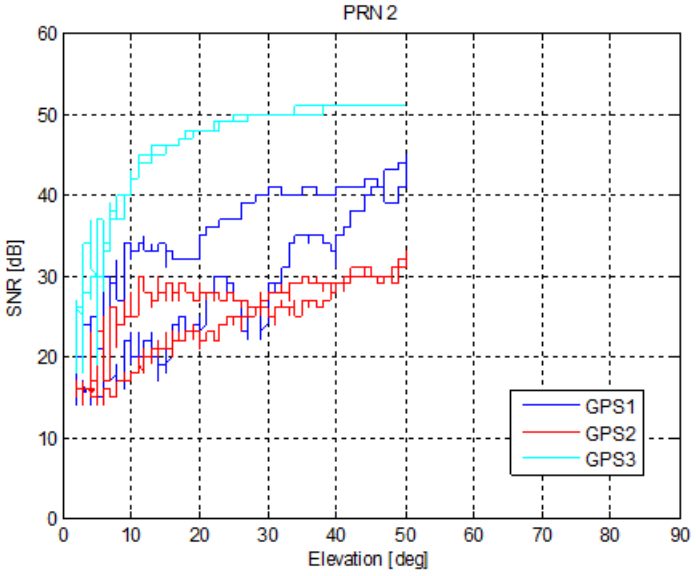

Figure 4.22: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 2

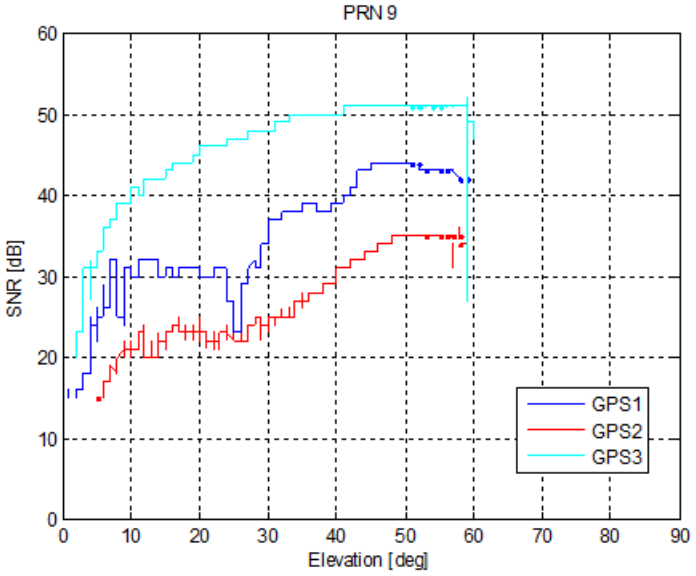

Figure 4.23: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 9

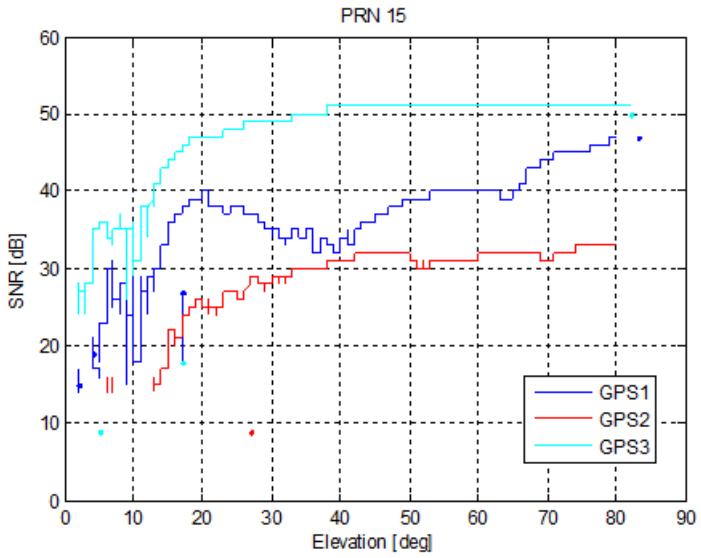

Figure 4.24: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 15

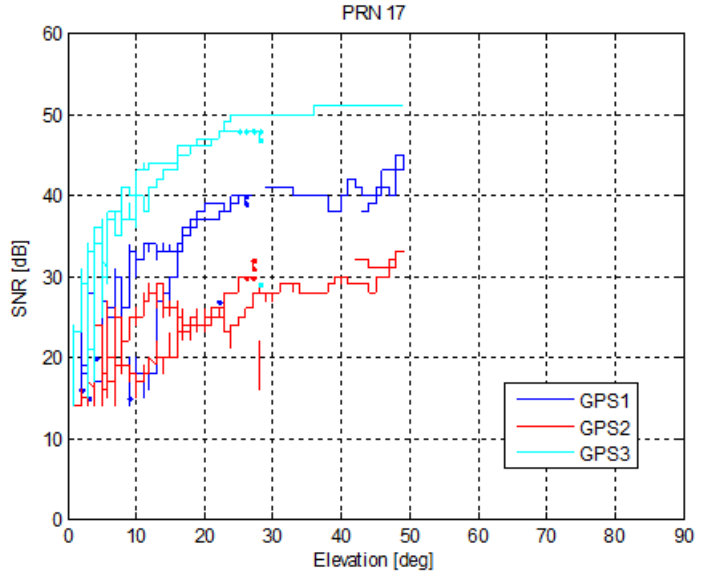

Figure 4.25: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 17

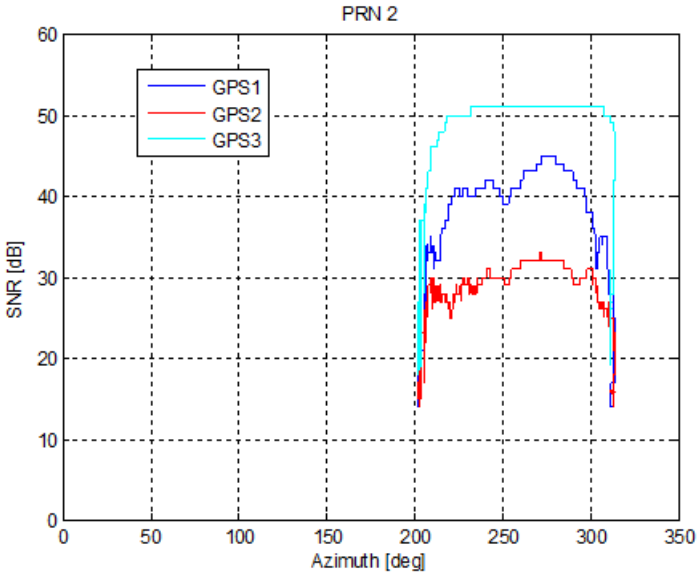

Figure 4.26: SNR (dB) with respect to azimuth (deg) for the 3 GPS modules for PRN 2

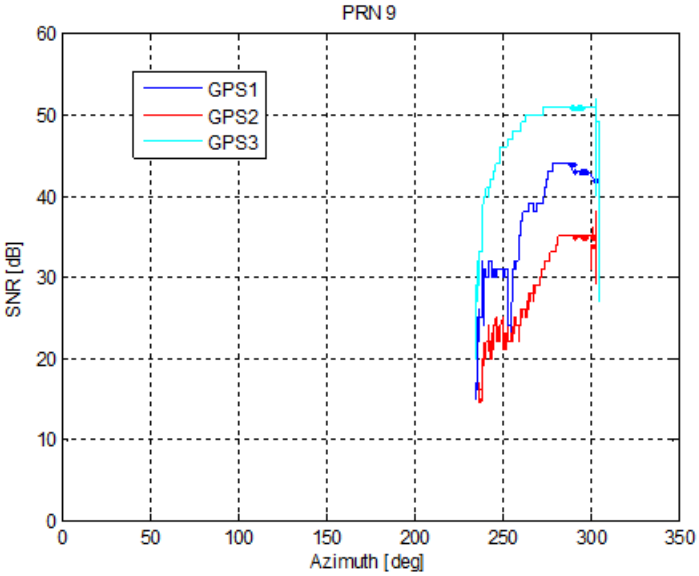

Figure 4.27: SNR (dB) with respect to azimuth (deg) for the 3 GPS modules for PRN 9

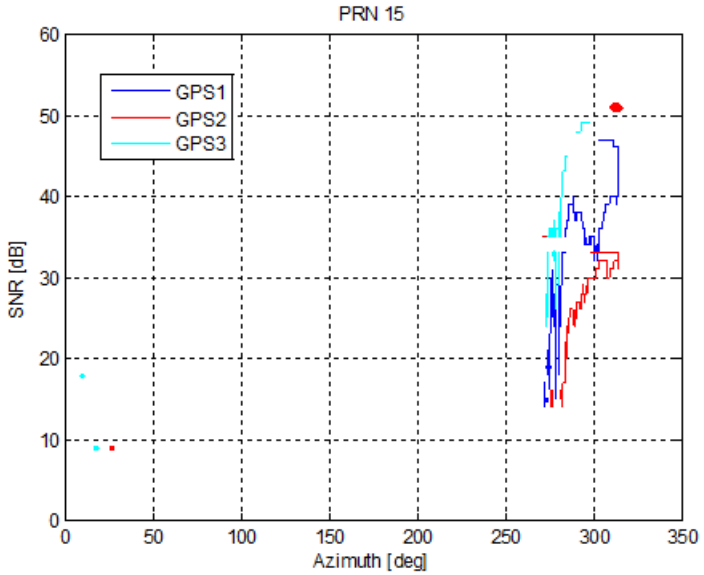

Figure 4.28: SNR (dB) with respect to azimuth (deg) for the 3 GPS modules for PRN 15

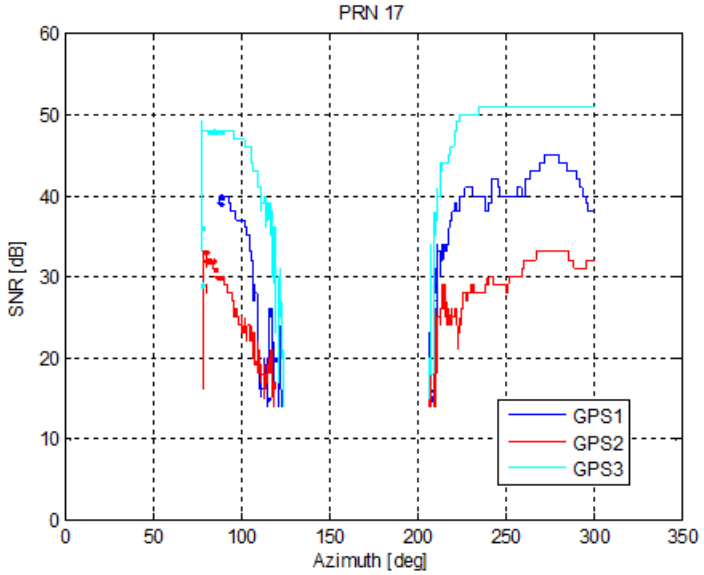

Figure 4.29: SNR (dB) with respect to azimuth (deg) for the 3 GPS modules for PRN 17

The SNR graphs with respect to the elevation show again a difference between GPS1 and GPS3 of 10 dB and a difference of 20 dB between GPS2 and GPS3, for elevation angle greater than 20°. The same behavior can

be observed for the SNR graphs with respect to the azimuth, taking into account that the non-perfectly symmetric configuration of the measurement setup can translate into asymmetry into the SNR response.

The following graphs represent the Position Dilution of Precision (PDOP), the Horizontal Dilution of Precision (HDOP) and the Vertical Dilution of Precision (VDOP). All these parameters give an indication of the quality that can be expected by the GPS computed position, based on the geometry of the received satellites. In fact values of DOP under the 6 can be used to obtain the position with a high level of confidence.

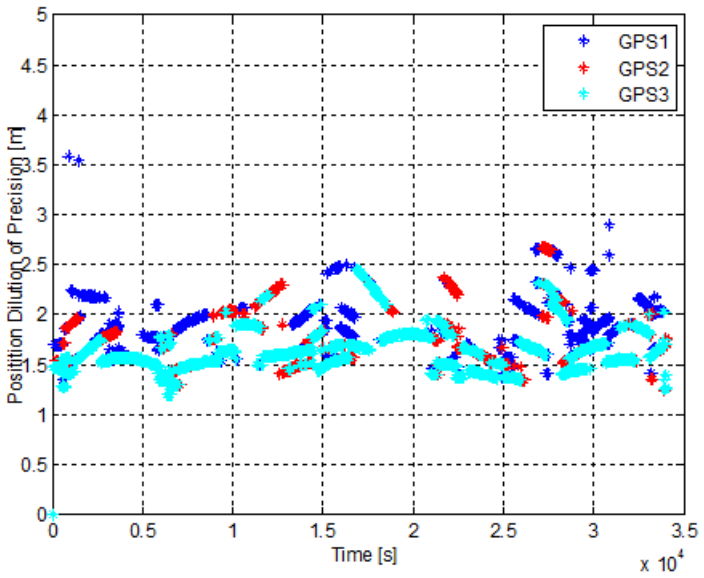

Figure 4.30: Position Dilution of Precision (PDOP) for the three GPS modules

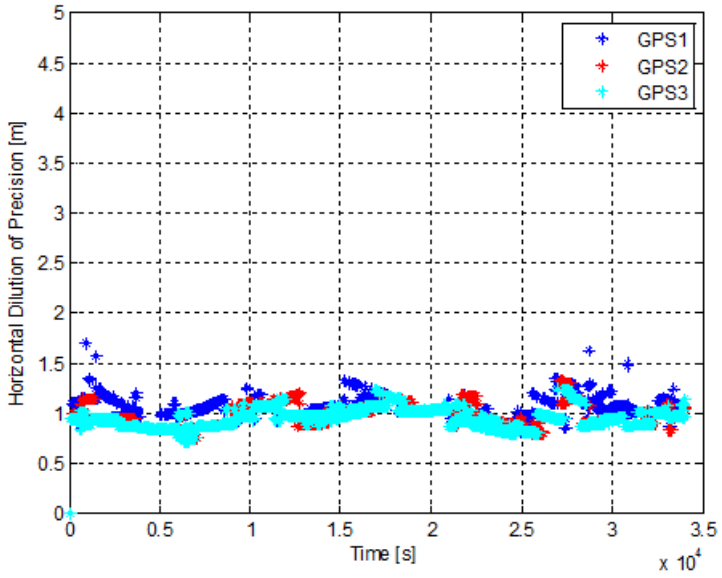

Figure 4.31: Horizontal Dilution of Precision (HDOP) for the three GPS modules

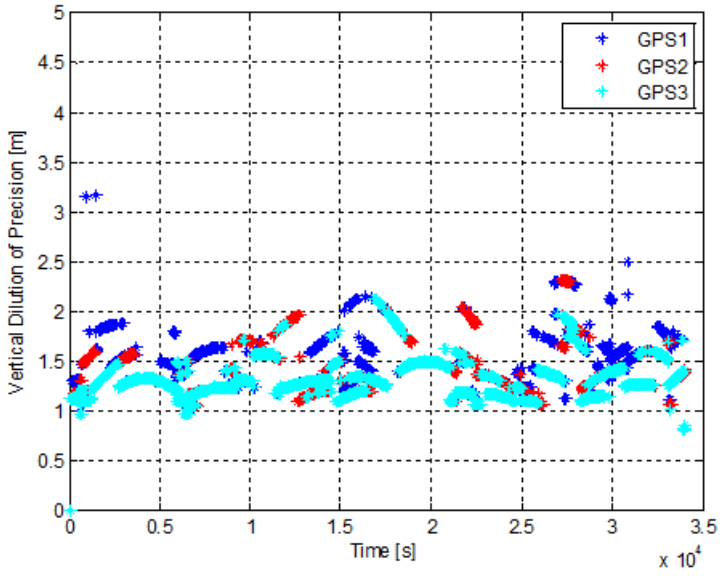

Figure 4.32: Vertical Dilution of Precision (VDOP) for the three GPS modules

The resulting signal quality is very good for the three GPS modules, with values in the range between 1 and 2.5, with just some few peaks. The GPS3

show, as expected, in 4.30, 4.31 and 4.32 the lowest DOP values. However the best performance in terms of signal quality is obtained for all the three receivers in the horizontal direction, as shown in 4.31.

The obtained results showed clearly the effect of water on the received GPS signal and demonstrated that the high sensitivity GPS modules used for the experiment have a strong sensibility to the variation of the surrounding dielectric material. Moreover the availability of a maximum of 20 dB of range in SNR difference between free space and water conditions seemed very promising in the perspective of the identification of different snow conditions, with dielectric characteristics between air and water.

4.2.3 Rice - Experiment I

The promising results obtained with the use of water leaded to the third experiment, which has been performed using as dielectric rice flour. In fact, according to several studies, the dielectric characteristics of rice flour approximate very well those of the dry snow, making possible to simulate quasi-real working conditions for the proposed system. The evaluated parameters were:

- the recorded position in terms of latitude, longitude and altitude
- the Signal-To-Noise Ratio(SNR) recorded by three GPS receivers for the available in view satellites
- the resulting dilution of precision in the horizontal and vertical direction (HDOP and VDOP)

The experiment has been performed placing the GPS receivers (GPS IT500) on the roof-top of the I+D building of the Universidad de Zaragoza. The data recording lasted for 24 hours, in two sessions, one performed in September the 5th and the second the following day. The measurements have been performed using two sacks containing 25 kg of rice flour from Harinas Polo, which size is reported in the following figure:

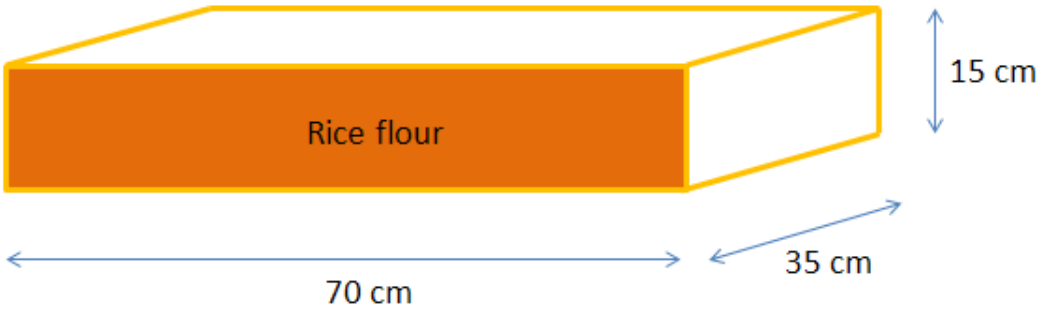

Figure 4.33: Dimensions of the rice flour sack

A commercial TDR (Campbell TDR100) together with a self-made three rod probe of 12 cm length were used to characterize the flour dielectric permittivity. The system is controlled via RS232 from a Panasonic Toughbook

by means of the free software PC-TDR provided by Campbell Scientific. A fast rise pulse is applied to the probe and the subsequent echo is recorded as a set of 512 points at 153 Gs/s. The signal stored is the average of 64 acquired waves and is saved as an ASCII file for further analysis. The system is calibrated to obtain the pulse round trip travel time with the probe in air. This time interval is obtained from the span between the two main maxima found in the TDR signal time derivative, as shown in Figure 2. A Chebychev class 2 low pass filter with 10 GHz cutoff frequency and 120 dB out of band suppression is applied to the TDR signal before differentiating, in order to remove unwanted noise that otherwise ruins the results. The same procedure is applied when the probe is immersed in a dielectric. A longer roundtrip time is then obtained. The square of the ratio of this two times is the real part of the effective dielectric permittivity.

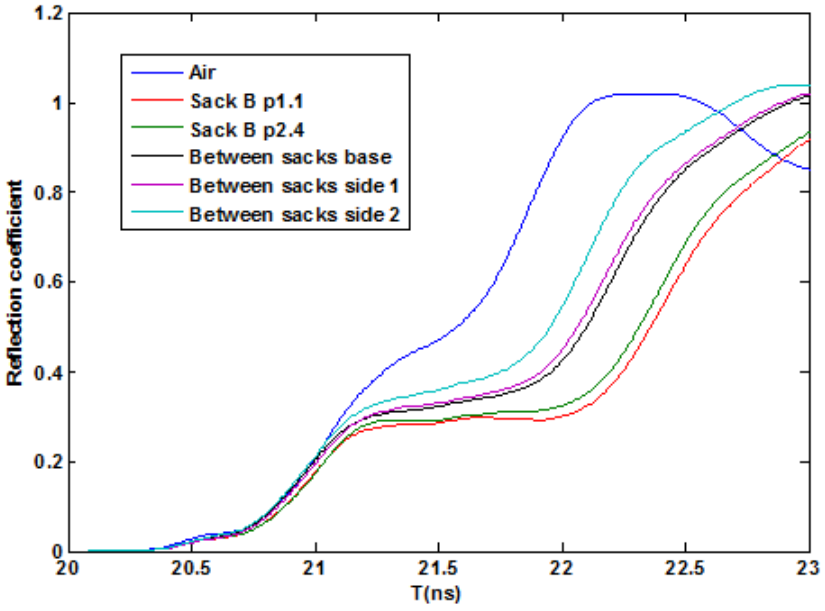

Figure 4.34: Smoothed TDR signal and its time derivative with the probe in air.

The calibration procedure is repeated four times. From the calibration data, a mean value of $0.819 \pm 0.011 ns$ is obtained for the travel time in air.

The TDR signal was subsequently measured with the probe inserted in each sack, named A and B, in two different points (p1), close to the base, p2, some cm towards the top) and also between both sacks in four different positions (base, side 1, side 2 and top) and finally below the sacks, 4.35.

Figure 4.35: Rice flour sacks with the used measurement points.

The permittivity was calculated with the procedure already described. The results are summarized in 4.1.

Data stored in file	ΔT (ns)	ϵ_r
sack A p11	1.415	2.99
sack A p12	1.409	2.96
sack A p21	1.389	2.88
sack A p22	1.383	2.85
sack B p11	1.422	3.01
sack B p12	1.422	3.01
sack B p13	1.422	3.01
sack B p14	1.422	3.01
sack B p21	1.383	2.84
sack B p22	1.383	2.84
sack B p23	1.376	2.82
sack B p24	1.376	2.82
between sacks base	1.226	2.24
between sacks side1	1.187	2.09
between sacks side2	1.096	1.79
between sacks top	1.089	1.76
under sack B p21	1.161	2.00
under sack B p22	1.148	1.96
under sack B p23	1.148	1.96
under sack B p24	1.142	1.94

Table 4.1: Measured data: round trip travel time with the probe inserted in different positions

The rice flour relative permittivity varies with the degree of powder compaction, as expected. Near the base of the sacks (p1) ϵ_r is very close to 3, decreasing to 2.83 in p2 where the floor is somewhat looser. The effective permittivity when measured between the sacks is affected in the same way by the amount of space occupied by the flour: the values are higher near the bases and decrease towards the top.

The three receivers were placed according to the configuration reported in figue 4.36:

The first GPS antenna has been placed directly on the roof floor and covered with one sack. Then the second GPS antenna has been placed over

Figure 4.36: Lateral view of the used measurement configurations. On the right the experiment with rice sacks and on the left, the same configuration, but with empty boxes

the first sack and covered itself with the second sack. The third GPS antenna has been placed on the top of the whole structure. The first day the three GPS receivers have been tested using an analogous configuration, made up with two empty card board boxes in order to dispose the same configuration but replacing the dielectric media with air. Such configuration has been used as reference, to evaluate the effect on each GPS receiver of the presence of the rice sack.

In the following are reported the graphs obtained processing the GPS output data with Matlab. The graphs are divided into two time intervals of 12 hours each, according to the orbital period of GPS satellites and to simplify data management. The reported graphs refer to the data obtained with the rice sacks configuration.

The following graphs report the latitude, longitude and altitude computed by the GPS receivers.

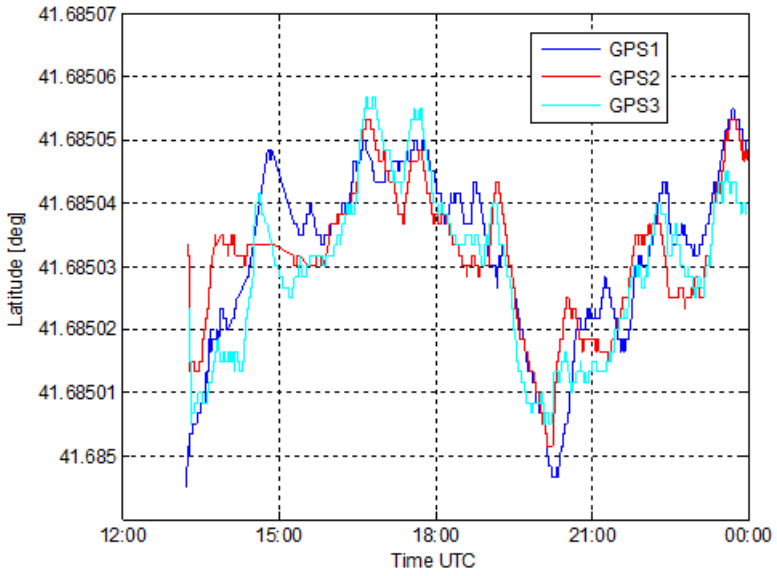

Figure 4.37: Comparison between the latitude (deg) with respect to time (UTC) recorded by the 3 GPS modules (day-time)

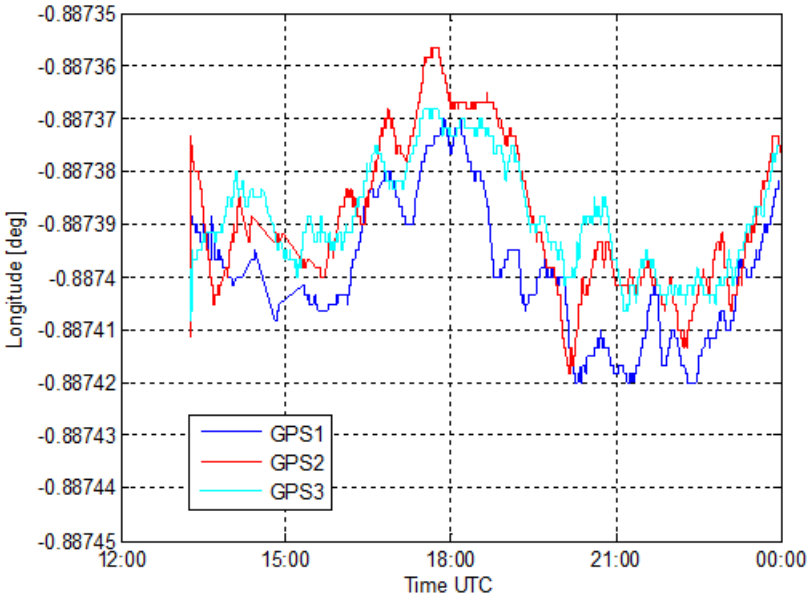

Figure 4.38: Comparison between the longitude (deg) with respect to time (UTC) recorded by the 3 GPS modules (day-time)

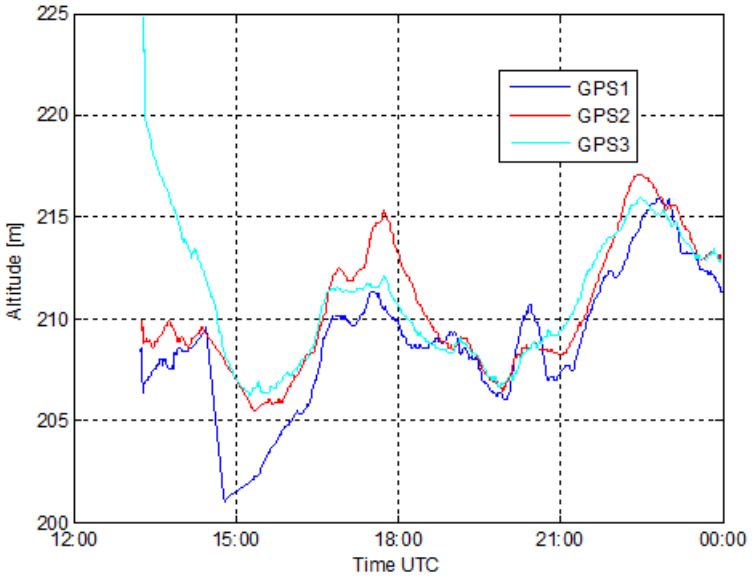

Figure 4.39: Comparison between the longitude (deg) with respect to time (UTC) recorded by the 3 GPS modules (day-time)

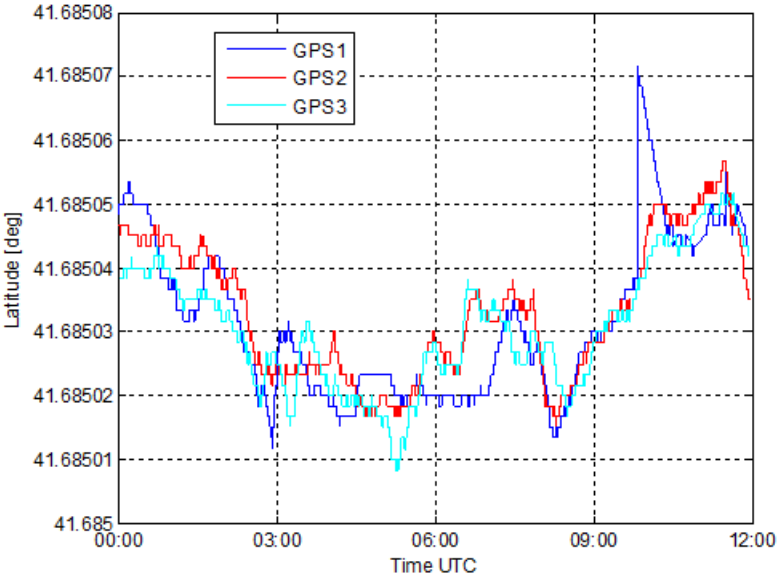

Figure 4.40: Comparison between the latitude (deg) with respect to time (UTC) recorded by the 3 GPS modules (night-time)

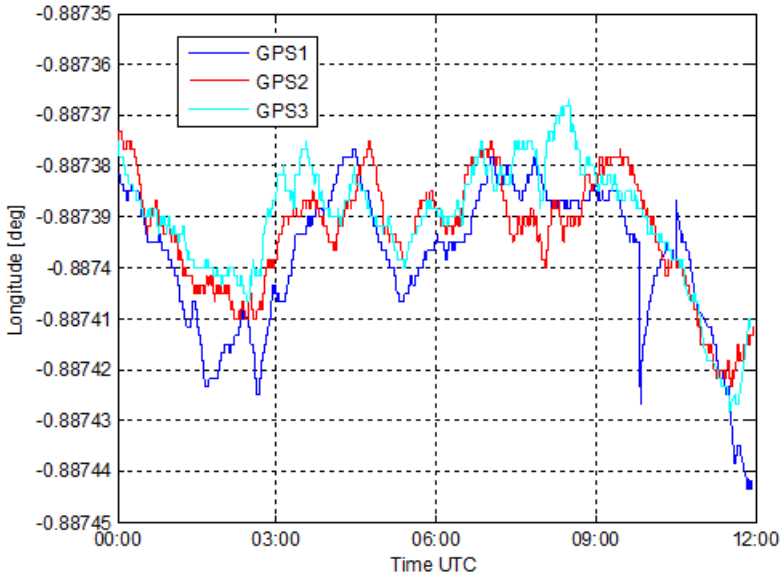

Figure 4.41: Comparison between the longitude (deg) with respect to time (UTC) recorded by the 3 GPS modules (night-time)

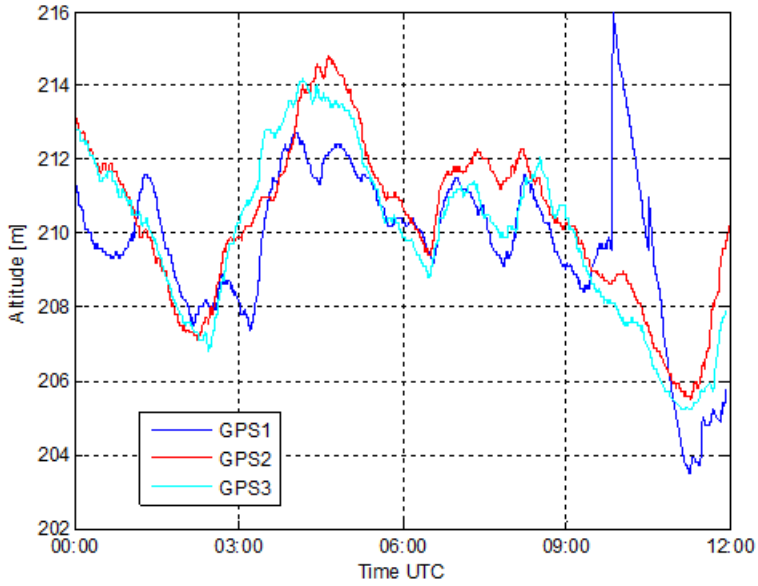

Figure 4.42: Comparison between the longitude (deg) with respect to time (UTC) recorded by the 3 GPS modules (night-time)

The evaluation of the computed positions shows that both day-time and night-time the three GPS receivers are in good agreement. The presence of some peaks in 4.40, 4.39, 4.42 and it is probably due to some temporary interference.

The following graphs report the comparison between latitude and longitude with respect to time recorded by each GPS antenna in the two different measurement days (with and without rice sacks).

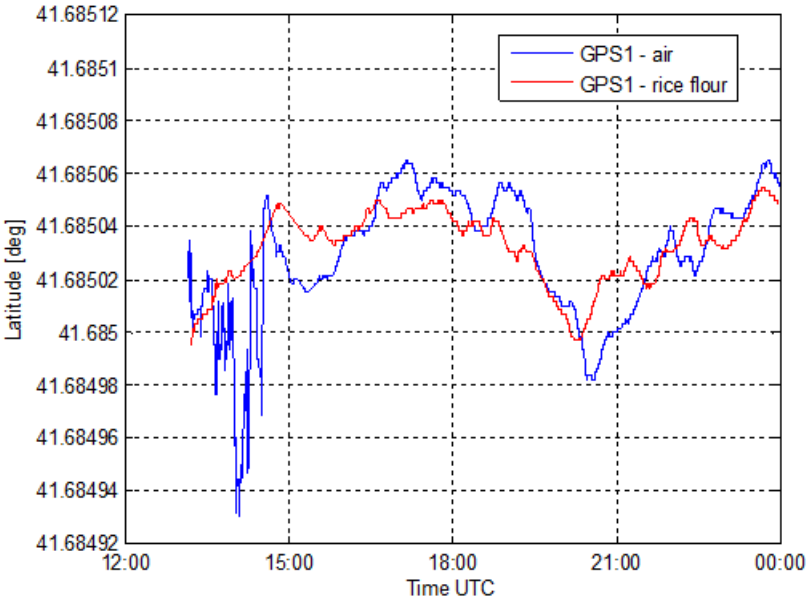

Figure 4.43: Latitude (dB) with respect to time (UTC) for the GPS1 (air and rice), (day-time)

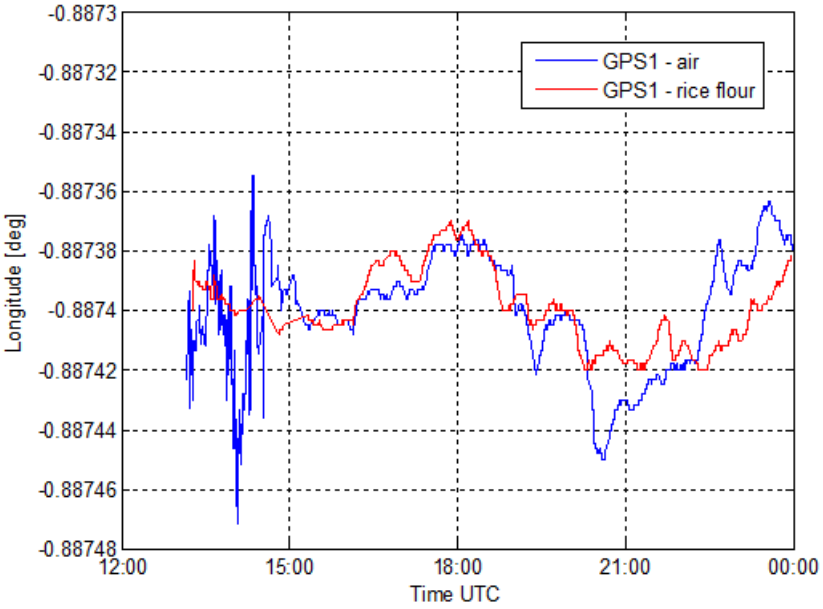

Figure 4.44: Longitude (dB) with respect to time (UTC) for the GPS1 (air and rice), (day-time)

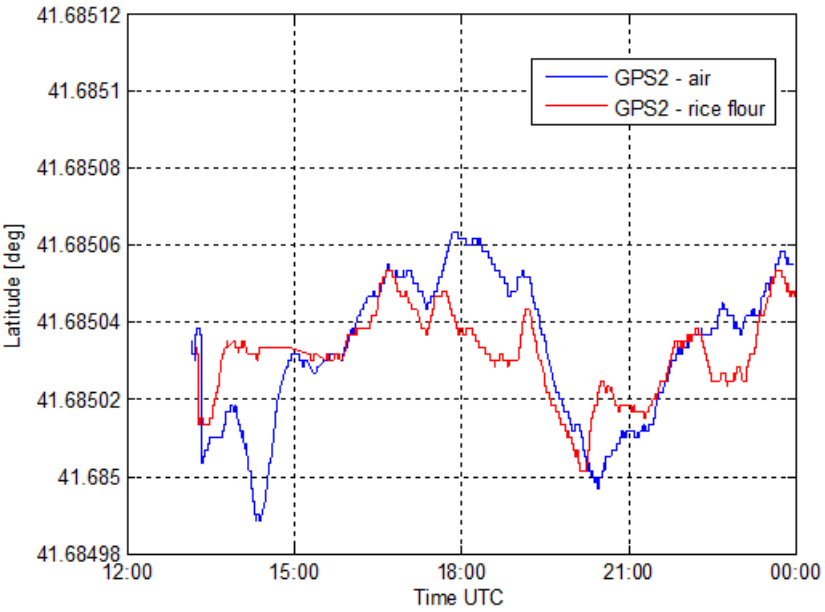

Figure 4.45: Latitude (dB) with respect to time (UTC) for the GPS2 (air and rice), (day-time)

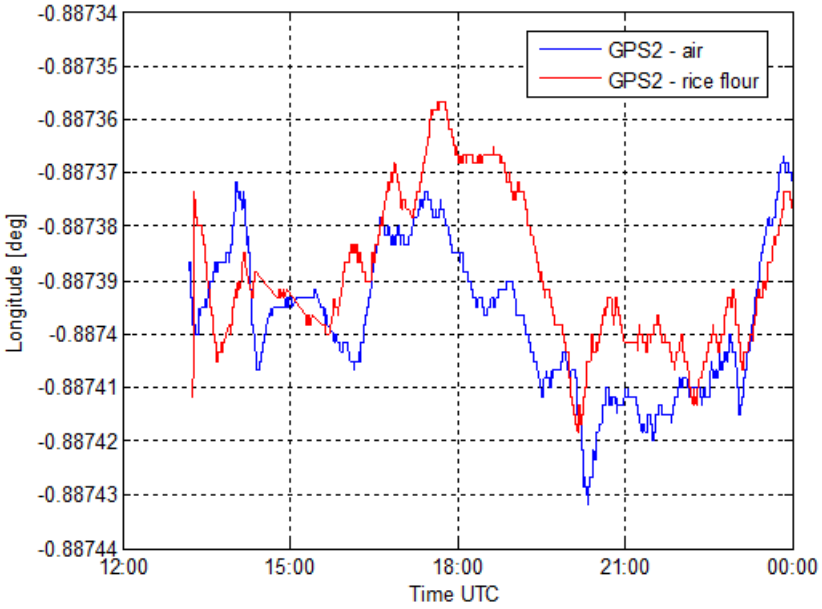

Figure 4.46: Longitude (dB) with respect to time (UTC) for the GPS2 (air and rice), (day-time)

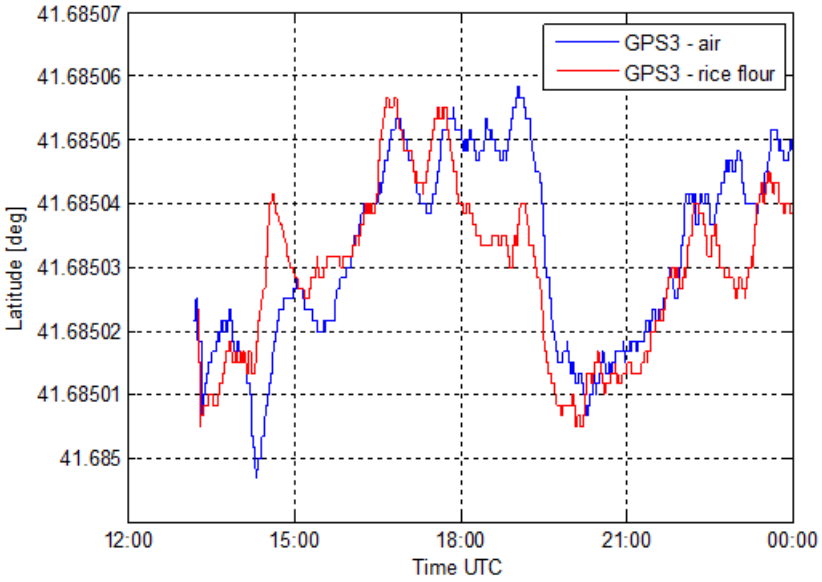

Figure 4.47: Latitude (dB) with respect to time (UTC) for the GPS3 (air and rice), (day-time)

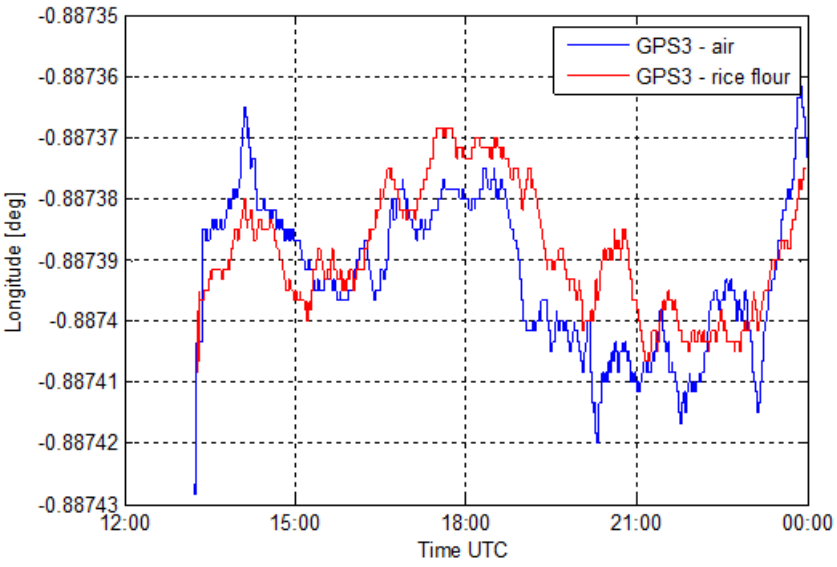

Figure 4.48: Longitude (dB) with respect to time (UTC) for the GPS3 (air and rice), (day-time)

The reported graphs show that the position obtained by the three GPS receivers, in both cases, is in good agreement. In fact the used receivers are able to obtain a consistent position even when the received signal is lower than in normal conditions. In 4.43 and 4.44 it is possible to observe peaks of noise related to the latitude and longitude obtained by GPS1 without rice sacks, probably associated to interference in the first three hours of signal acquisition. The same behavior can be observed for the other receivers, but in a lighter form, as in 4.48 at 14:00 UTC time.

The following graphs represent the SNR variation of the different in view satellites recorded by each GPS module as a function of time. The data are again divided into two intervals of time of 12 hours each. The reported graphs include the satellites in view for the longest period and with the best performance in terms of SNR: PRN 1, 11, 13 and 23 day-time and PRN 13, 23 and 28 night-time.

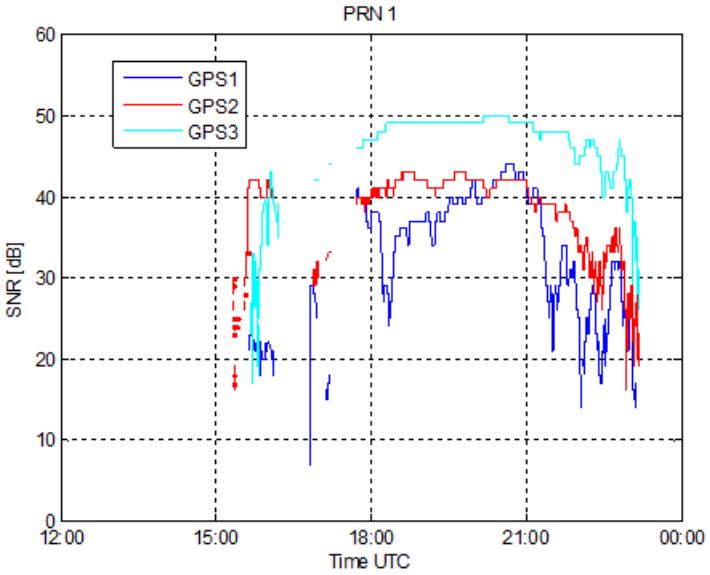

Figure 4.49: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 1 (day-time)

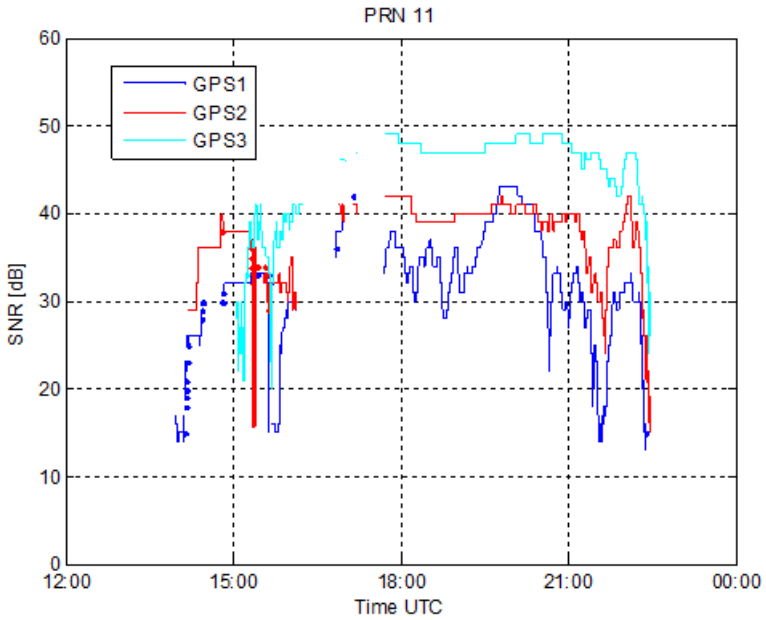

Figure 4.50: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 11 (day-time)

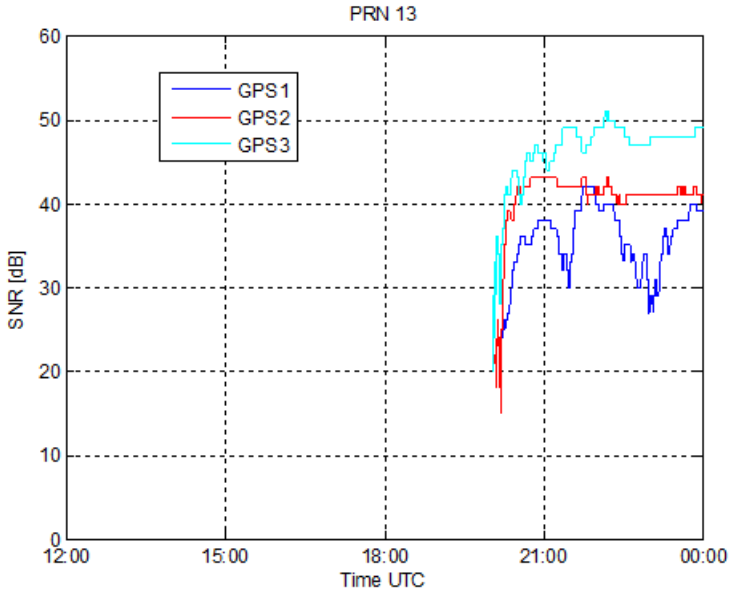

Figure 4.51: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 13 (day-time)

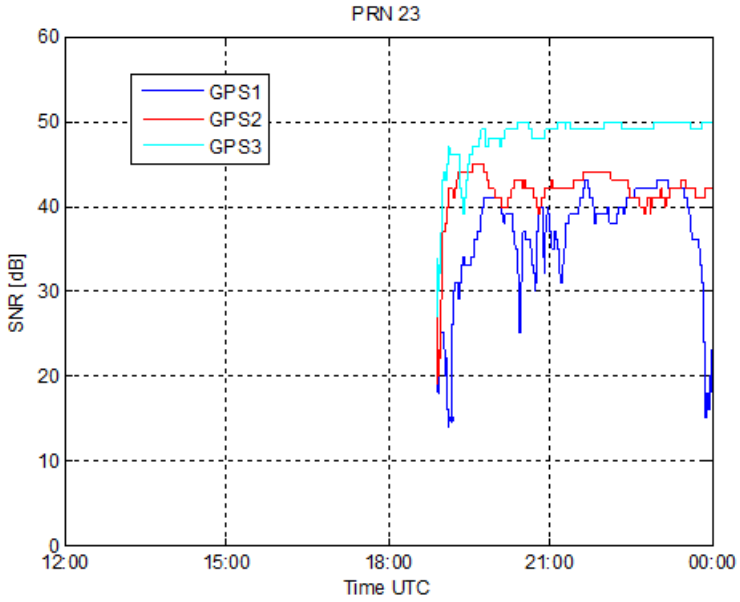

Figure 4.52: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 23 (day-time)

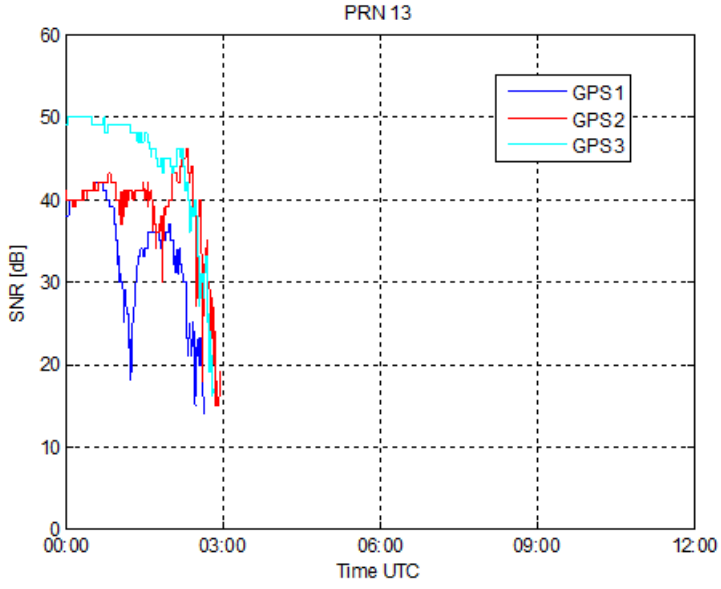

Figure 4.53: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 13 (night-time)

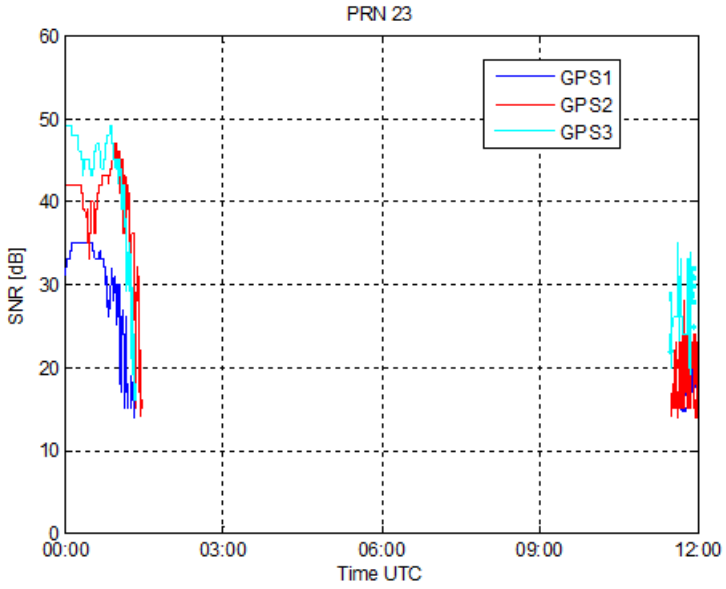

Figure 4.54: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 23 (night-time)

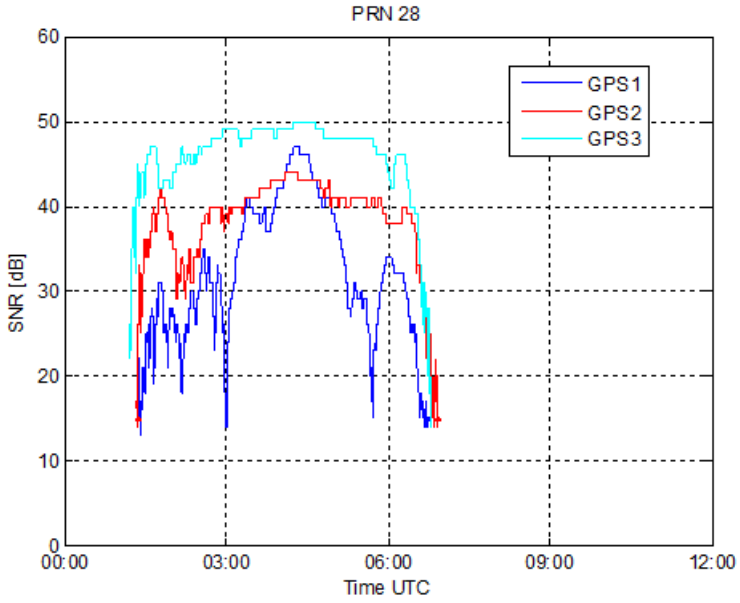

Figure 4.55: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 28 (night-time)

The reported graphs for the SNR obtained for each GPS modules show the effect of rice: the resulting loss for all the satellites in view is around 8

dB for GPS2 (under one rice sack) and around 15 dB for GPS1 (under two rice sacks). The difference in SNR between GPS1 and GPS3 reaches peaks of 20 dB in 4.50, 4.51, 4.53, 4.54 and 4.55. The general behavior of the curves shows an almost constant difference in SNR between thee receivers, according reasonably to their position in the experimental setup. The presence of some inversion in the performance between GPS1 and GPS2, as in 4.49 (20:00 UTC time), 4.51 (20:00 UTC time) and 4.55 (05:00 UTC time), can be due to possible temporary interferences.

The following graphs report the SNR variation recorded by the GPS modules for satellites with PRN 1,11,13 and 23 (day-time) and PRN 13, 23 and 28 (night-time) with respect to the elevation (degrees) and the azimuth (degrees).

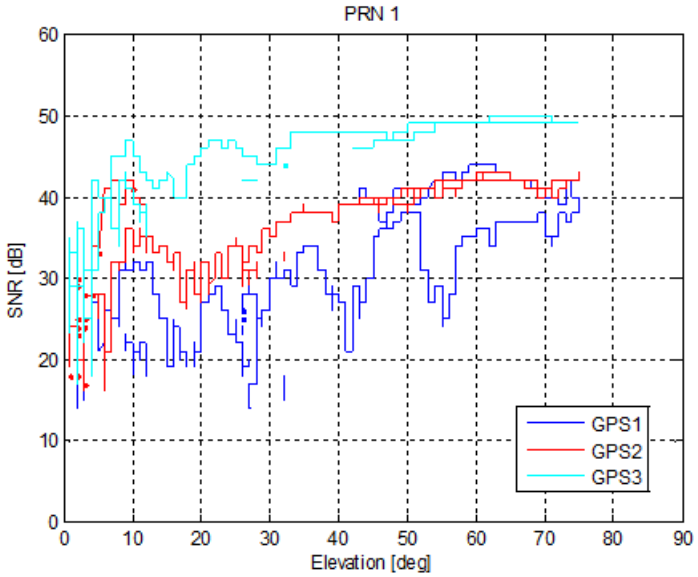

Figure 4.56: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 1 (day-time)

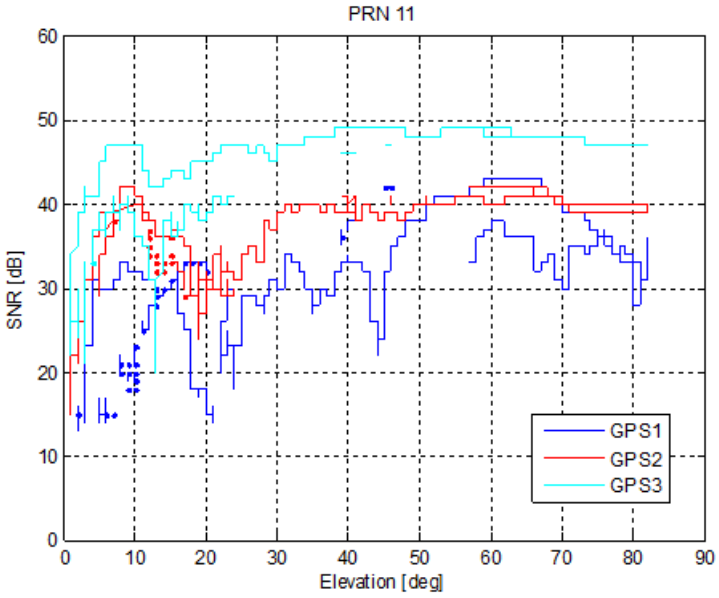

Figure 4.57: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 11 (day-time)

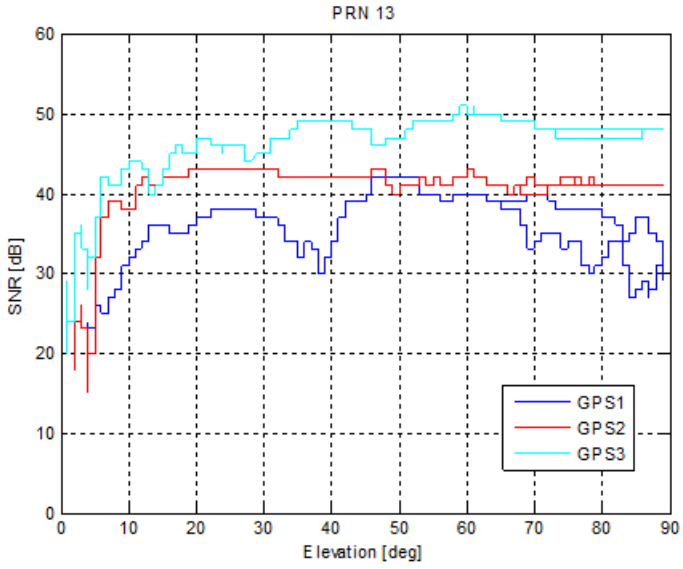

Figure 4.58: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 13 (day-time)

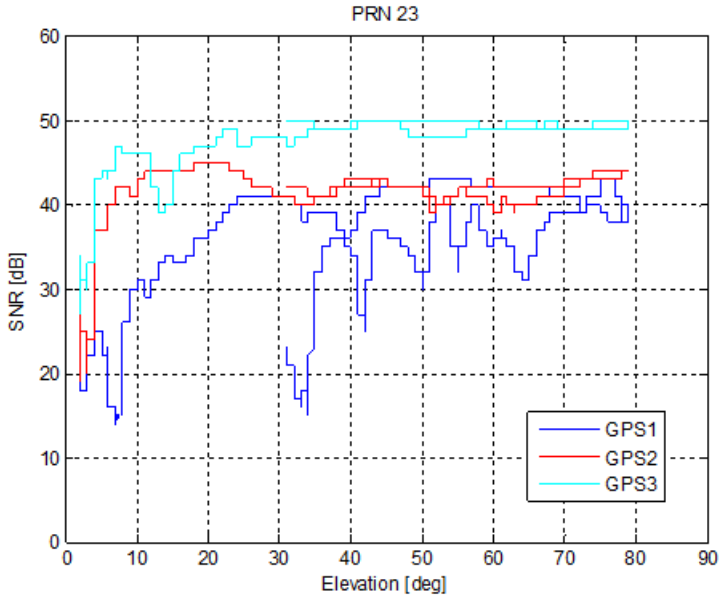

Figure 4.59: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 23 (day-time)

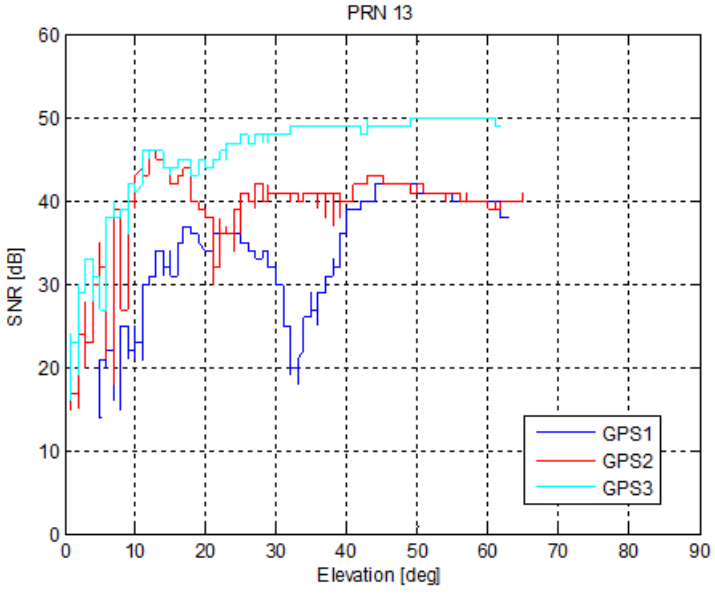

Figure 4.60: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 13 (night-time)

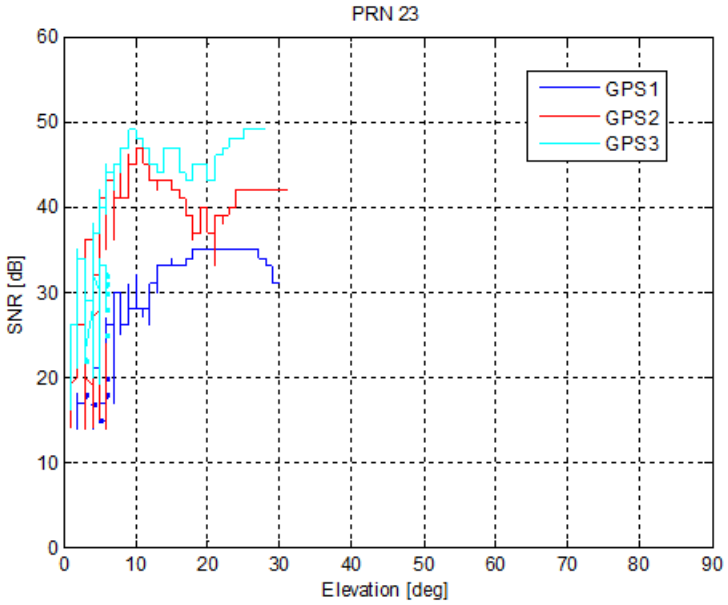

Figure 4.61: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 23 (night-time)

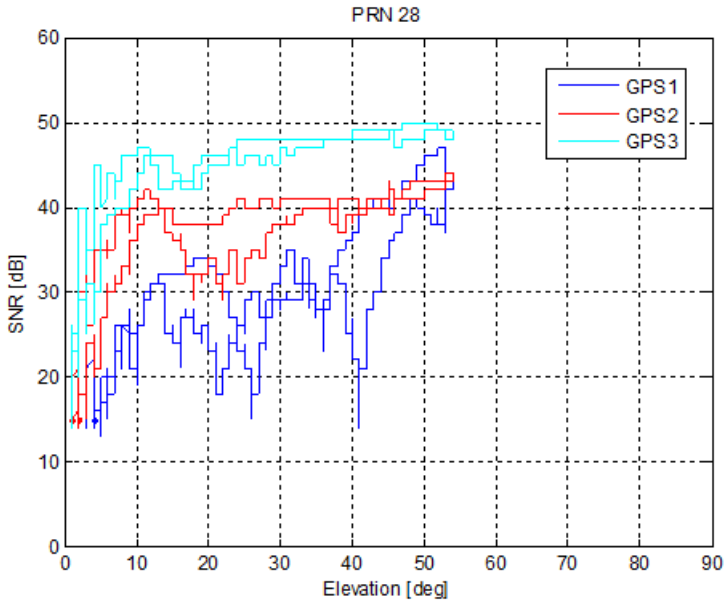

Figure 4.62: SNR (dB) with respect to elevation (deg) for the 3 GPS modules for PRN 28 (night-time)

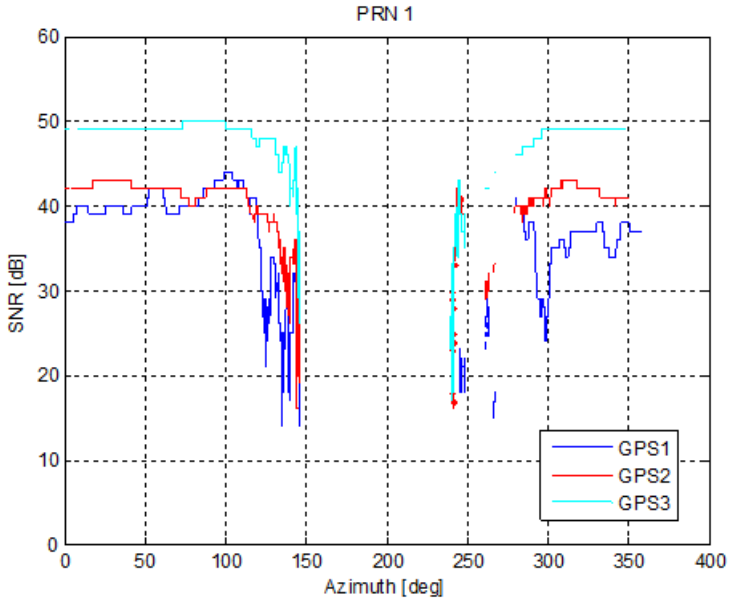

Figure 4.63: SNR (dB) with respect to azimuth (deg) for the 3 GPS modules for PRN 1 (day-time)

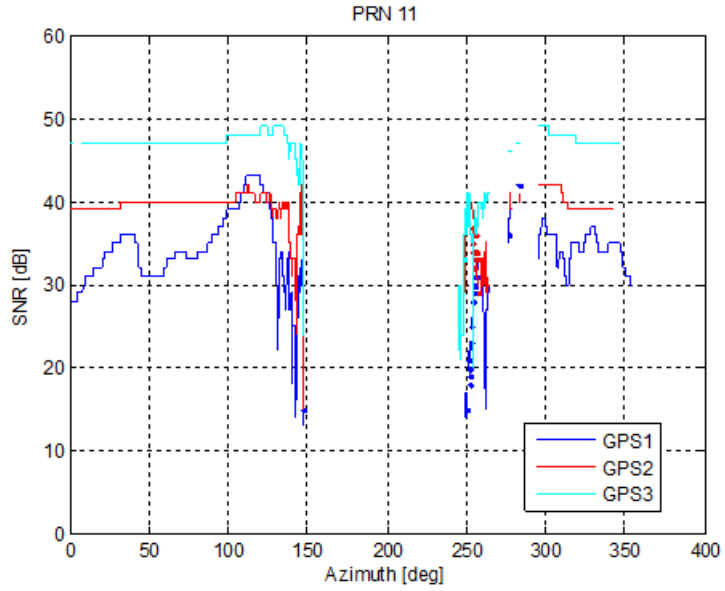

Figure 4.64: SNR (dB) with respect to azimuth (deg) for the 3 GPS modules for PRN 11 (day-time)

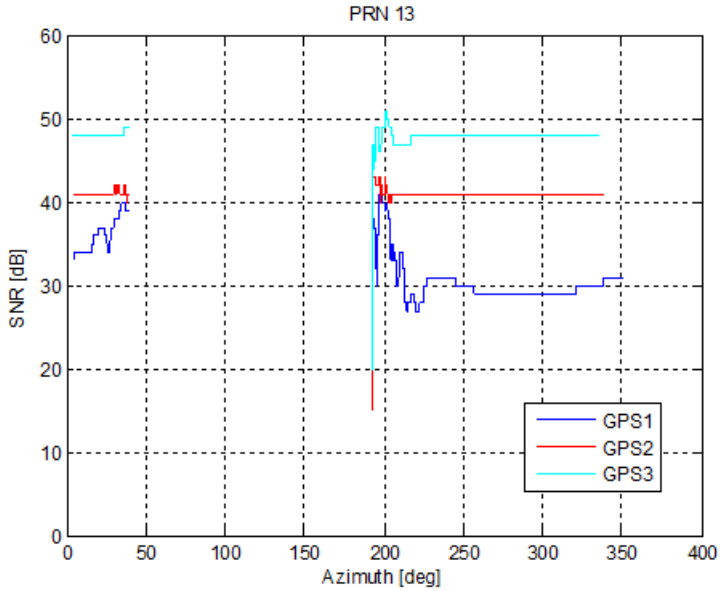

Figure 4.65: SNR (dB) with respect to azimuth (deg) for the 3 GPS modules for PRN 13 (day-time)

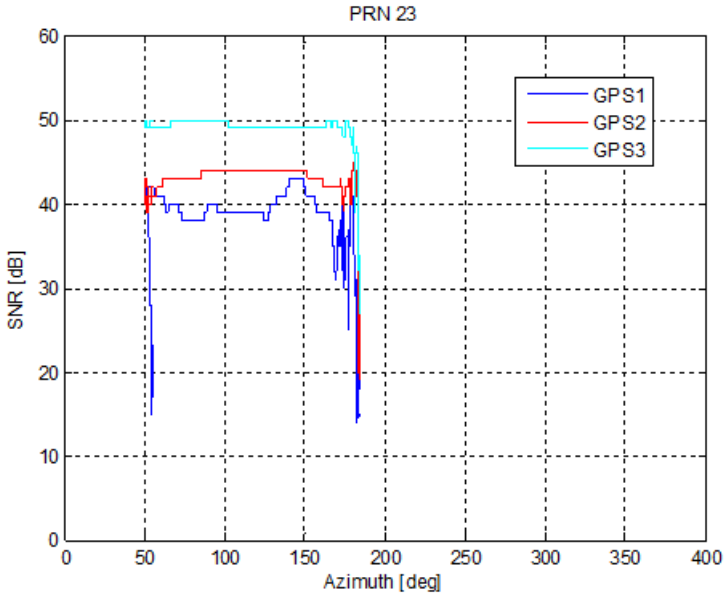

Figure 4.66: SNR (dB) with respect to azimuth (deg) for the 3 GPS modules for PRN 23 (day-time)

The SNR graphs with respect to the elevation show again a difference between GPS2 and GPS3 of 8 dB and a difference of 15 dB between GPS3

and GPS1, for elevation angle greater than 20°. The same behavior can be observed for the SNR graphs with respect to the azimuth, taking into account that the non-perfectly symmetric configuration of the measurement setup (GPS1 and GPS3 have been placed both close to the extreme of the respective sacks) can translate into asymmetry into the SNR response. The graphs respect to elevation and azimuth show the same peaks of 20 dB of SNR difference between GPS1 and GPS3, as in 4.56, 4.57, 4.60, 4.62, 4.63 and 4.65.

In order to assess the effect of the rice on the performance of each single GPS receiver, in the following graphs is reported the comparison between the SNR obtained by the same GPS receiver in the two different measurement days (with and without rice). The following graphs reports the obtained SNR with respect to time, for the three GPS receivers, recorded during day-time (satellites with PRN 1, 11, 13 and 23).

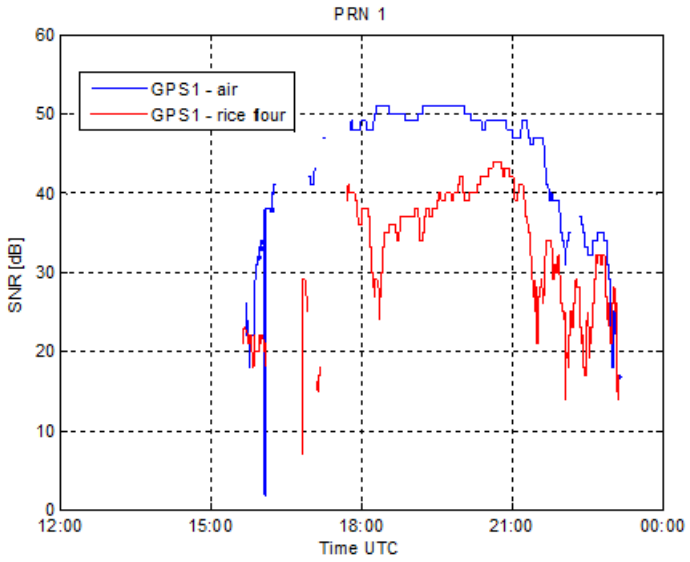

Figure 4.67: SNR (dB) with respect to time (UTC) for the GPS1 (air and rice) for PRN 1 (day-time)

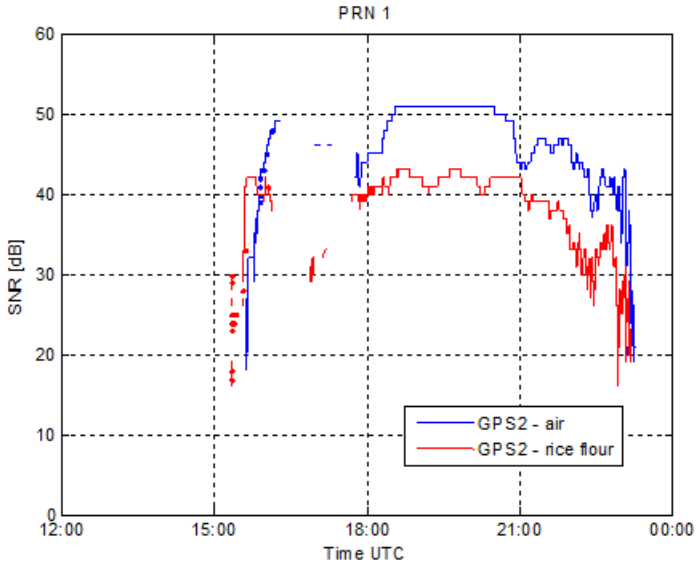

Figure 4.68: SNR (dB) with respect to time (UTC) for the GPS2 (air and rice) for PRN 1 (day-time)

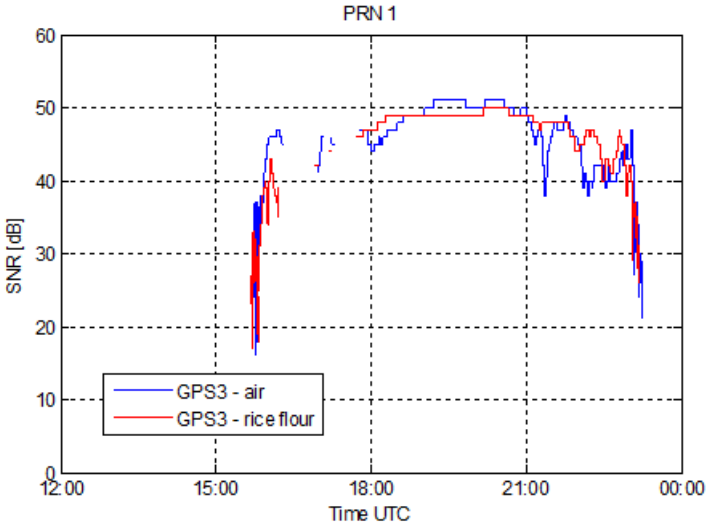

Figure 4.69: SNR (dB) with respect to time (UTC) for the GPS3 (air and rice) for PRN 1 (day-time)

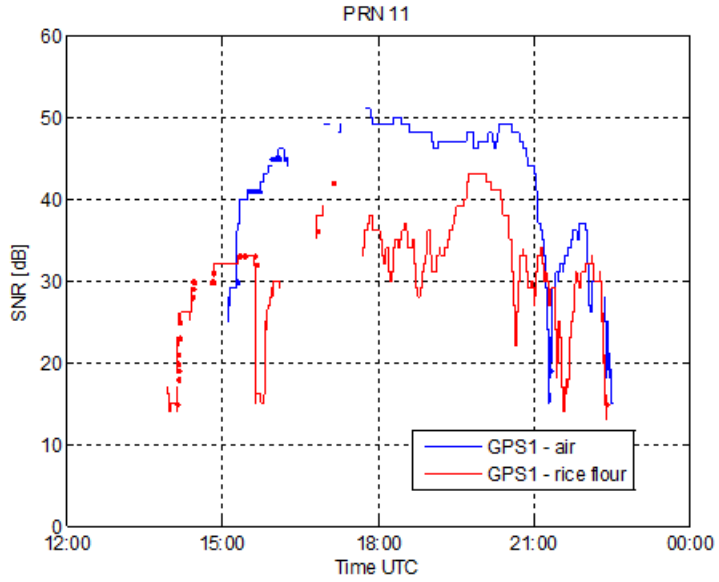

Figure 4.70: SNR (dB) with respect to time (UTC) for the GPS1 (air and rice) for PRN 11 (day-time)

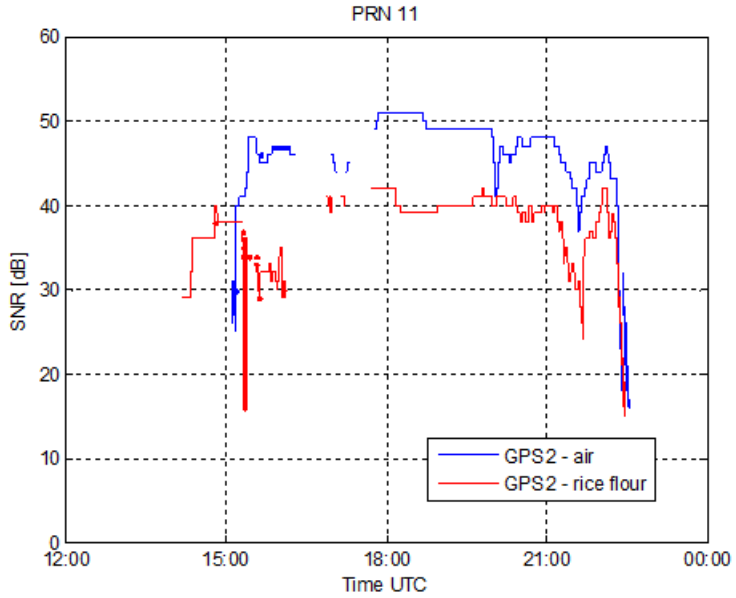

Figure 4.71: SNR (dB) with respect to time (UTC) for the GPS2 (air and rice) for PRN 11 (day-time)

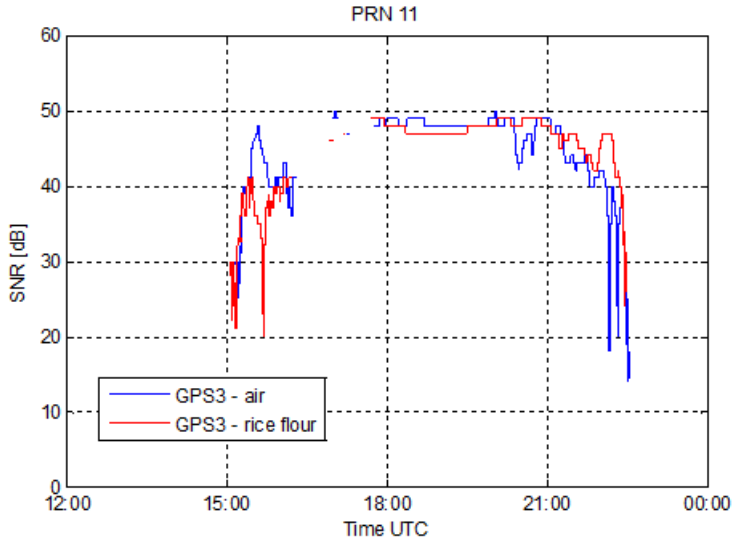

Figure 4.72: SNR (dB) with respect to time (UTC) for the GPS3 (air and rice) for PRN 11 (day-time)

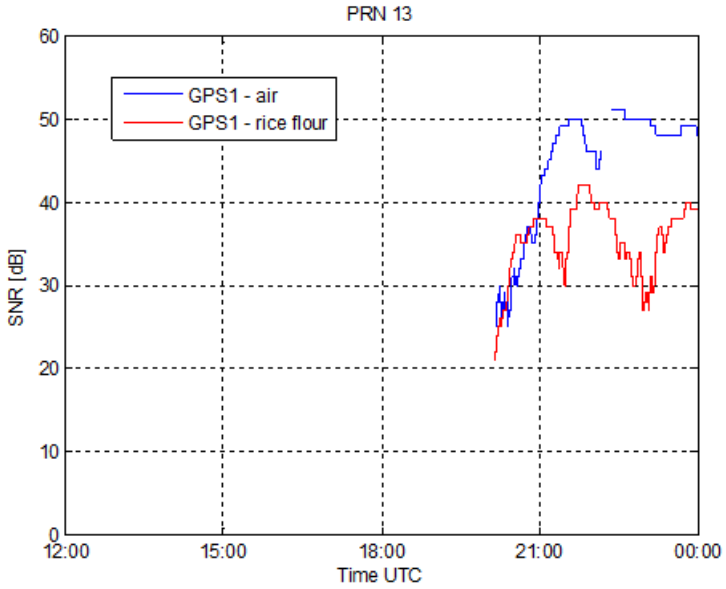

Figure 4.73: SNR (dB) with respect to time (UTC) for the GPS1 (air and rice) for PRN 13 (day-time)

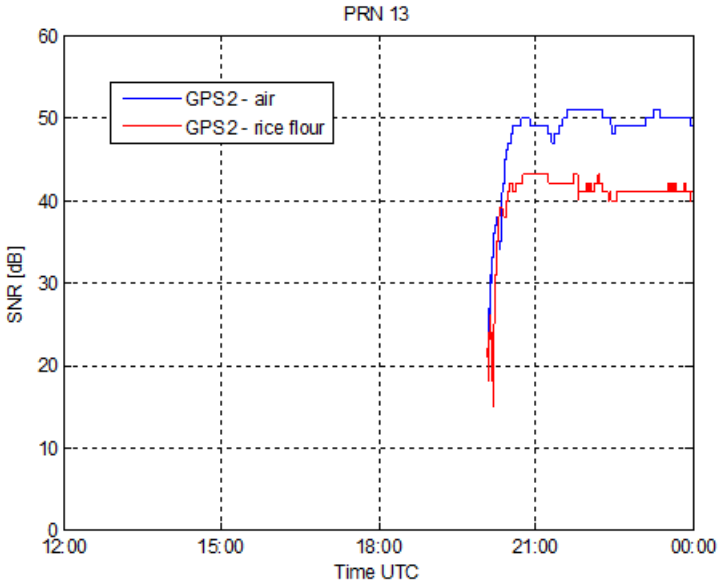

Figure 4.74: SNR (dB) with respect to time (UTC) for the GPS2 (air and rice) for PRN 13 (day-time)

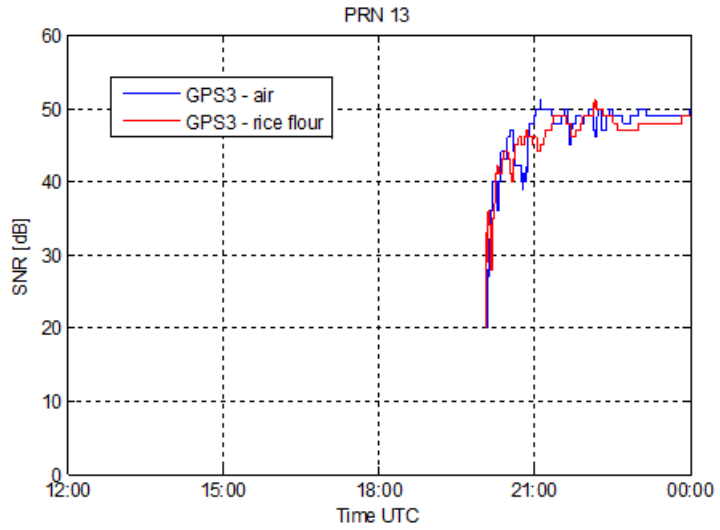

Figure 4.75: SNR (dB) with respect to time (UTC) for the GPS3 (air and rice) for PRN 13 (day-time)

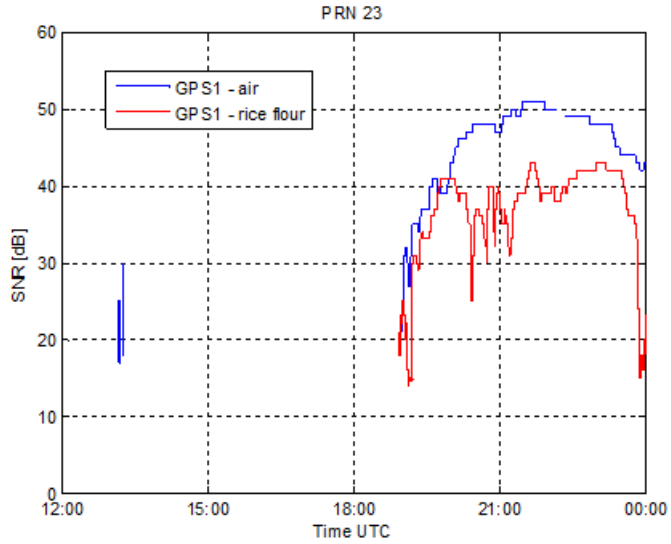

Figure 4.76: SNR (dB) with respect to time (UTC) for the GPS1 (air and rice) for PRN 23 (day-time)

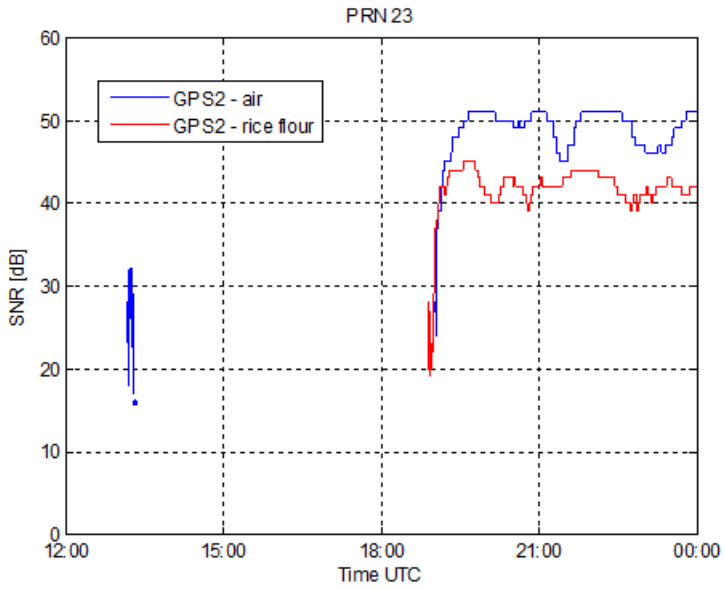

Figure 4.77: SNR (dB) with respect to time (UTC) for the GPS2 (air and rice) for PRN 23 (day-time)

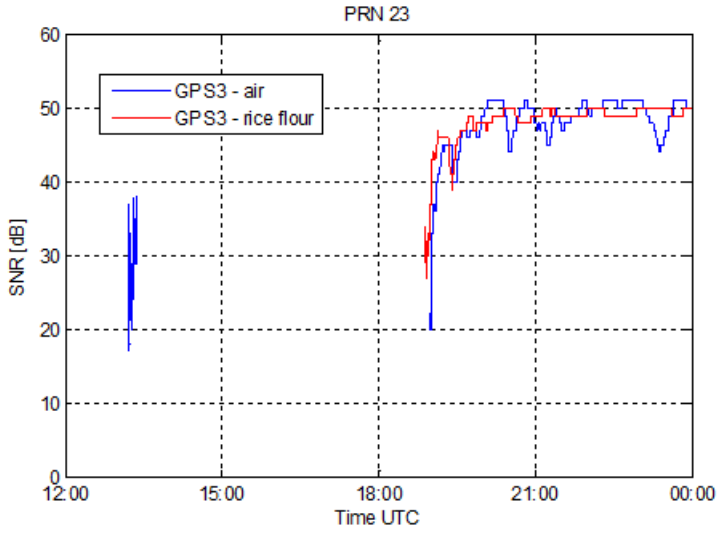

Figure 4.78: SNR (dB) with respect to time (UTC) for the GPS3 (air and rice) for PRN 23 (day-time)

The reported graphs show clearly the effect of rice on the received signal. In fact both GPS1 and GPS2 lose around 10 dB in presence of rice, whereas the performance of GPS3 are almost unchanged, meaning that the presence of the underlying rice layer does not affect the receiver.

The following graphs represent the Position Dilution of Precision (PDOP), the Horizontal Dilution of Precision (HDOP) and the Vertical Dilution of Precision (VDOP). All these parameters give an indication of the quality that can be expected by the GPS computed position, based on the geometry of the received satellites. In fact values of DOP under the 6 can be used to obtain the position with a high level of confidence.

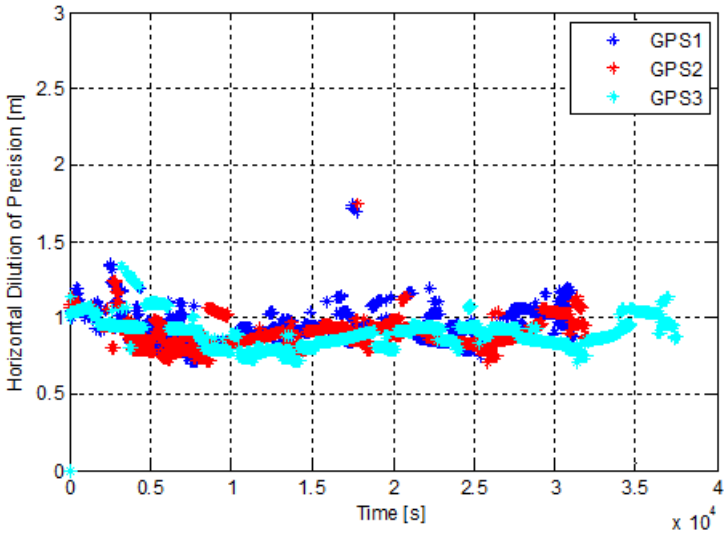

Figure 4.79: Horizontal Dilution of Precision (HDOP) for the three GPS modules (day-time)

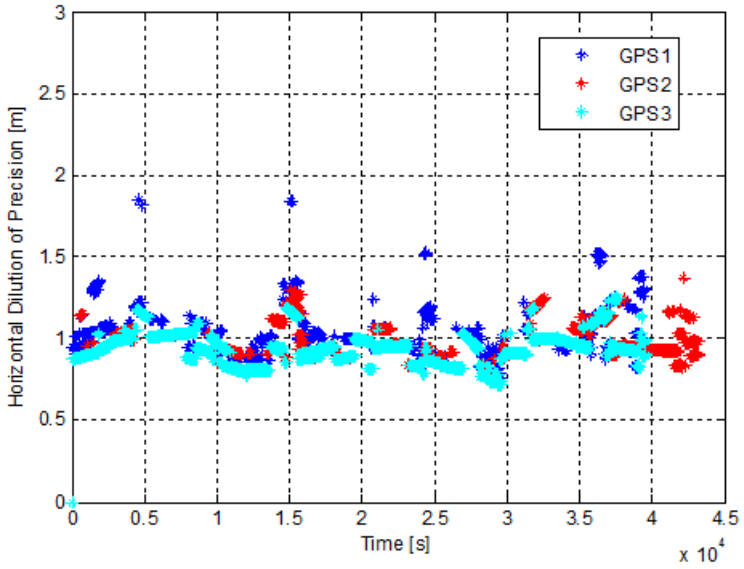

Figure 4.80: Horizontal Dilution of Precision (HDOP) for the three GPS modules (night-time)

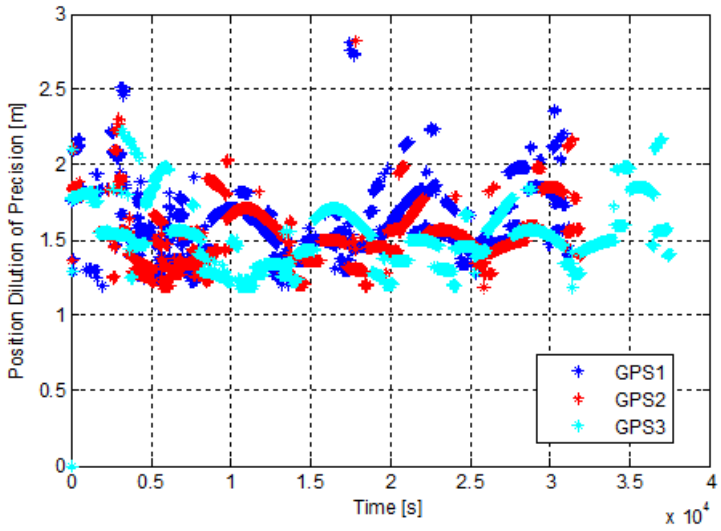

Figure 4.81: Position Dilution of Precision (PDOP) for the three GPS modules (day-time)

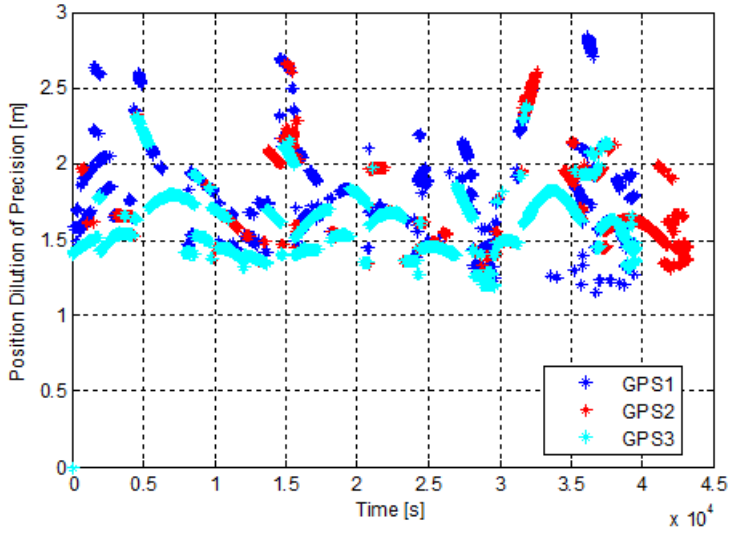

Figure 4.82: Position Dilution of Precision (PDOP) for the three GPS modules (night-time)

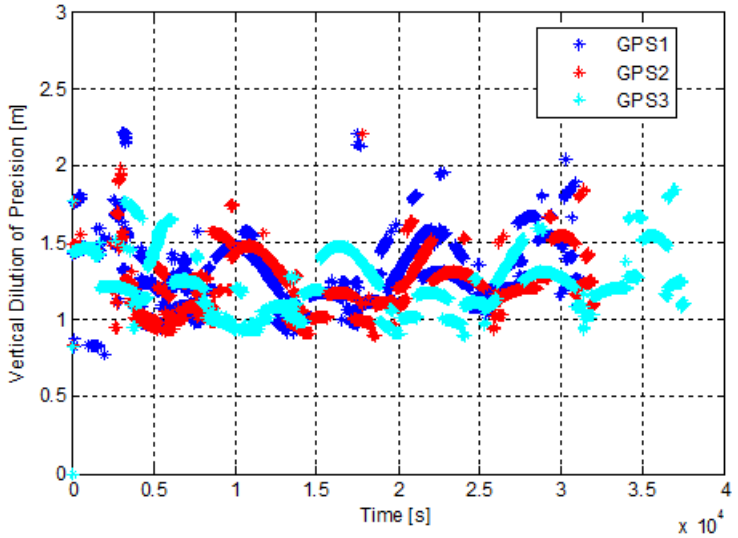

Figure 4.83: Vertical Dilution of Precision (VDOP) for the three GPS modules (day-time)

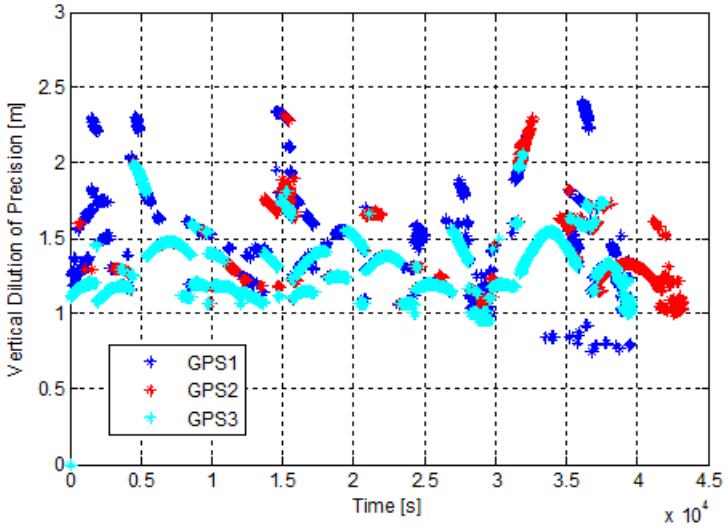

Figure 4.84: Vertical Dilution of Precision (VDOP) for the three GPS modules (night-time)

The resulting signal quality is very good for the three GPS modules, with values in the range between 1 and 2.5, with just some few peaks. The GPS3 shows, as expected the lowest DOP values both day-time and night-time. However the best performance in terms of signal quality is obtained for all the three receivers in the horizontal direction, as shown in 4.79 and 4.80.

The obtained results show clearly the effect of the rice on the received GPS signal and demonstrate that the high sensitivity GPS modules used for the experiment have a sensibility to the variation of the surrounding dielectric material, even if its dielectric characteristics are less strong than the water one. The similarities between the rice and dry snow dielectric characteristics make this result promising in the perspective of identifying different snow wetness conditions.

4.2.4 Rice - Experiment II a

The analysis of the capability of the GPS signal to detect different dielectric medium has been further deepen by performing an additional experiment, adding a rice sack in order to obtain the following configuration.

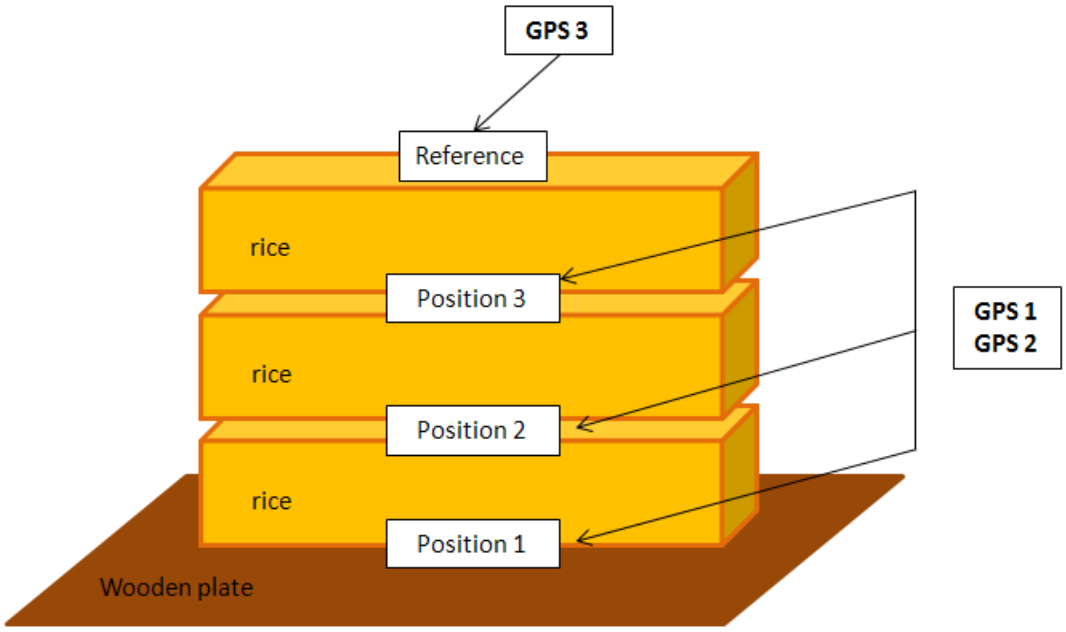

Figure 4.85: View of the used measurement configuration.

The experiment has been performed placing one the GPS antennas (GPS3) on top of the whole structure as reference and moving the others antennas every two hours from position 1 to position 3, as reported 4.85. Such configuration allowed to evaluate in a more precise way the effect of each rice sack. In fact both Position 2 and Position 3 have the same boundaries conditions and the only difference is the presence of an additional rice layer. Moreover the antennas were placed in the middle of the rice sack in order to improve the symmetry of the whole system. The data recording lasted for 6 hours.

In the following are reported the graphs obtained processing the GPS output data with Matlab. The graphs are divided in order to show on the same graph the reference antenna and one of the antennas placed in the three

different positions, in order to evaluate the difference on the data introduced by the rice layers.

The following graphs report the latitude, longitude and altitude computed by the GPS receivers.

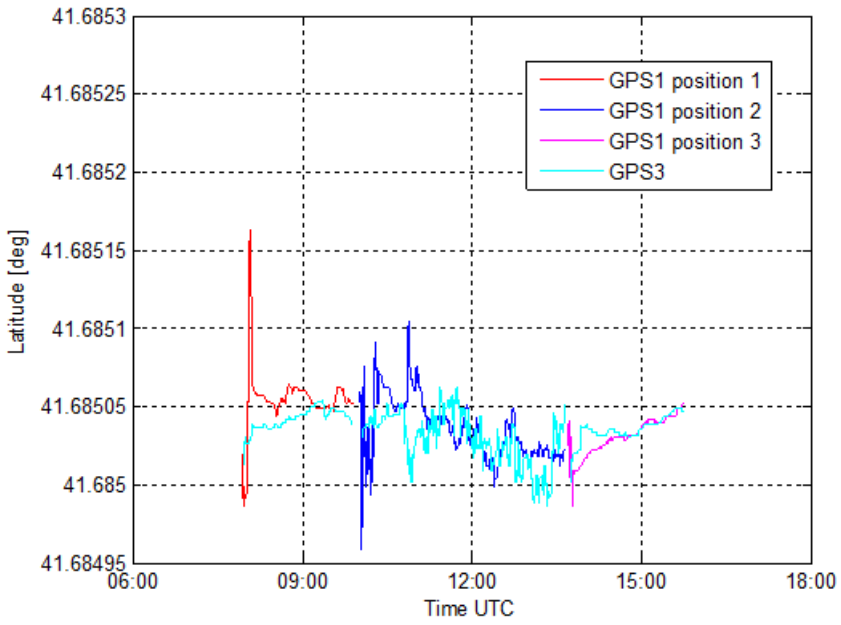

Figure 4.86: Comparison between the latitude (deg) with respect to time (UTC) recorded by GPS1 (in three different positions) and GPS3 (reference position)

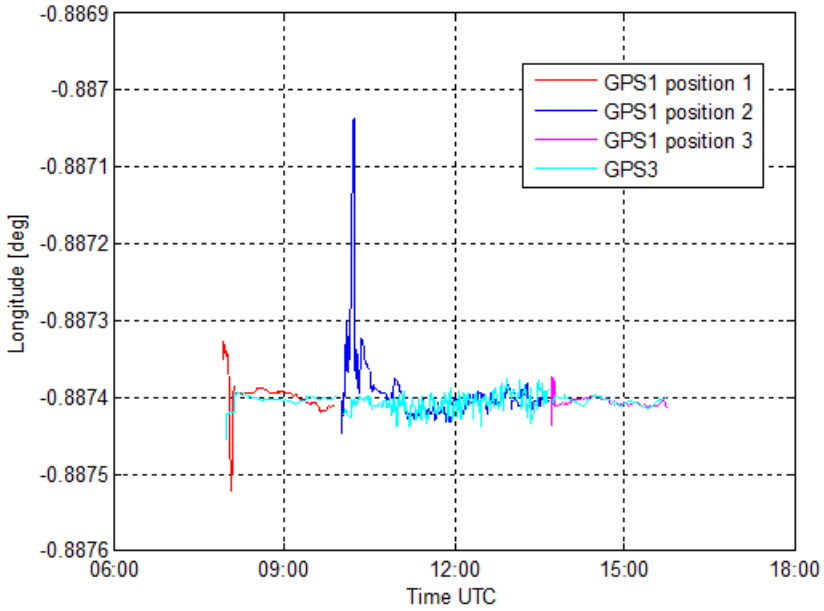

Figure 4.87: Comparison between the longitude (deg) with respect to time (UTC) recorded by GPS1 (in three different positions) and GPS3 (reference position)

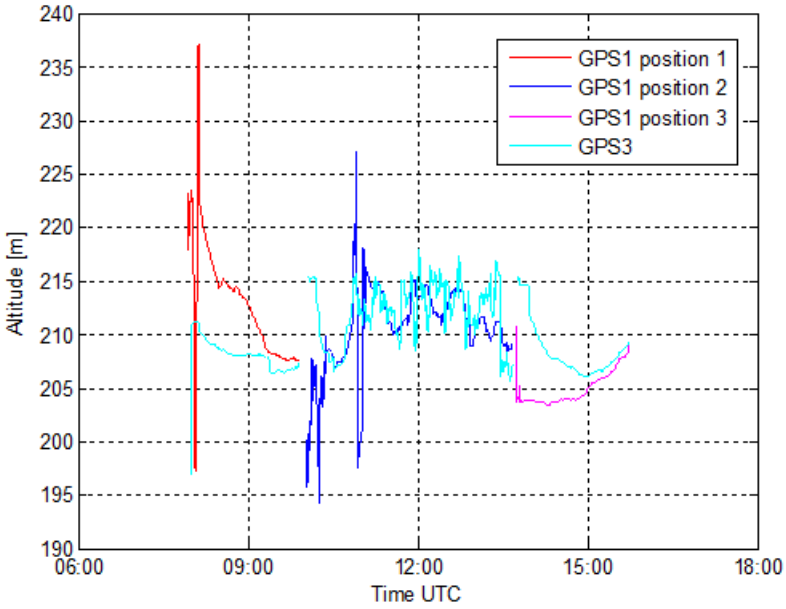

Figure 4.88: Comparison between the altitude (meters) with respect to time (UTC) recorded by GPS1 (in three different positions) and GPS3 (reference position))

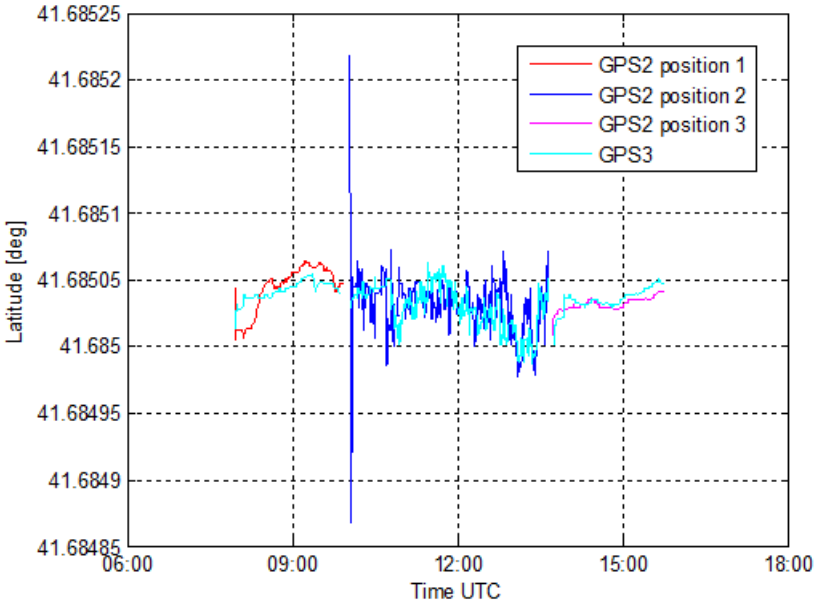

Figure 4.89: Comparison between the latitude (deg) with respect to time (UTC) recorded by GPS2 (in three different positions) and GPS3 (reference position)

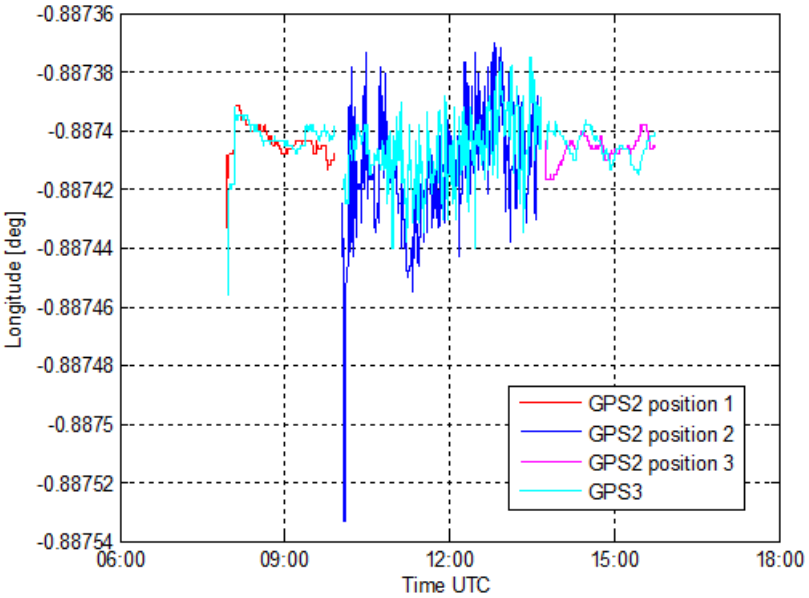

Figure 4.90: Comparison between the longitude (deg) with respect to time (UTC) recorded by GPS2 (in three different positions) and GPS3 (reference position)

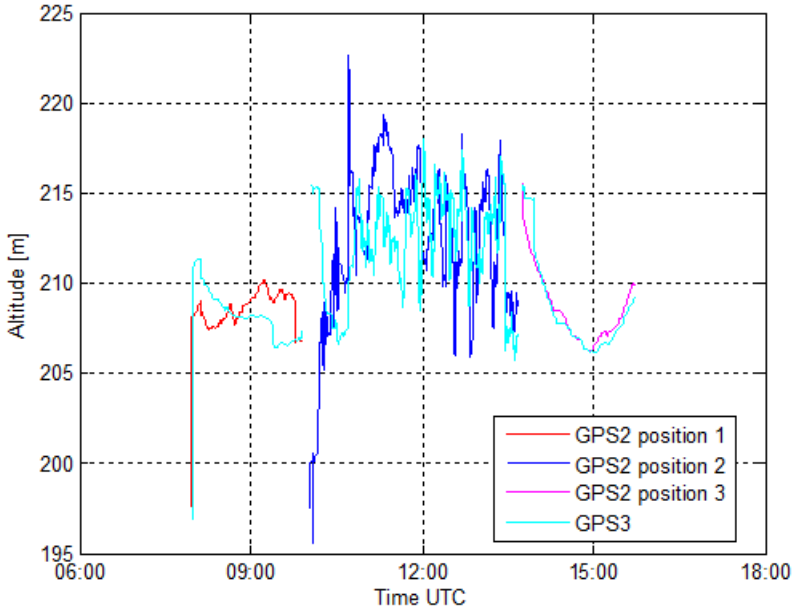

Figure 4.91: Comparison between the altitude (meters) with respect to time (UTC) recorded by GPS2 (in three different positions) and GPS3 (reference position))

The evaluation of the computed positions shows for both GPS1 and GPS2 the presence of high peaks due to some anomalous interference. Nevertheless the agreement between the reference antenna (GPS3) and the other antennas is good independently from the position and consequently from the rice layers.

The following graphs represent the SNR variation of the different in view satellites recorded by each GPS module as a function of time, first for position 1 and then for position 2. The data referred to position 3 are missing, as in that interval of the time the satellites in view were different, making difficult the comparison. The reported graphs include the satellites in view for the longest period and with the best performance in terms of SNR: PRN 25, 29 and 31.

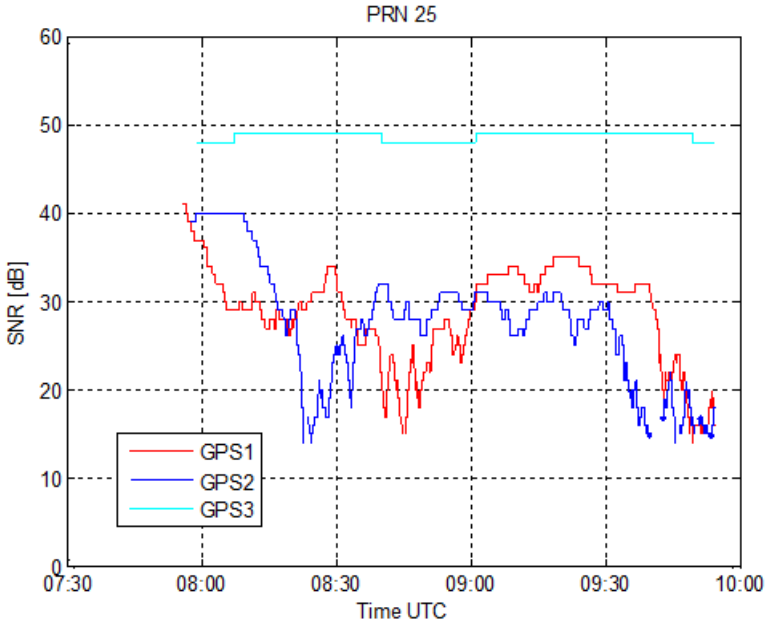

Figure 4.92: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 25 (position 1)

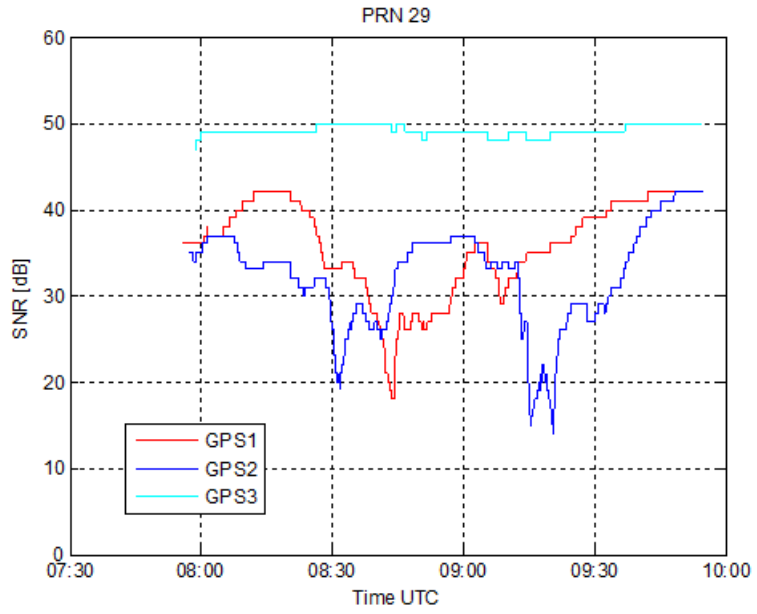

Figure 4.93: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 29 (position 1)

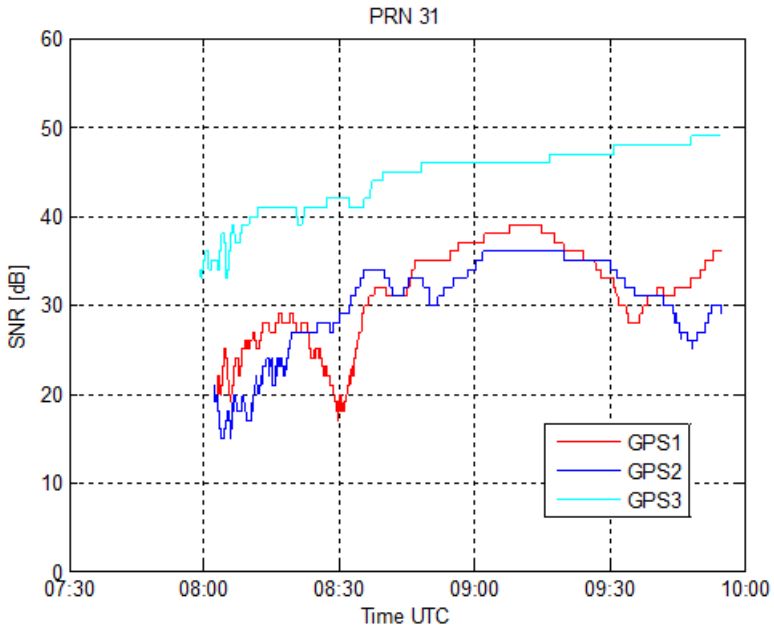

Figure 4.94: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 31 (position 1)

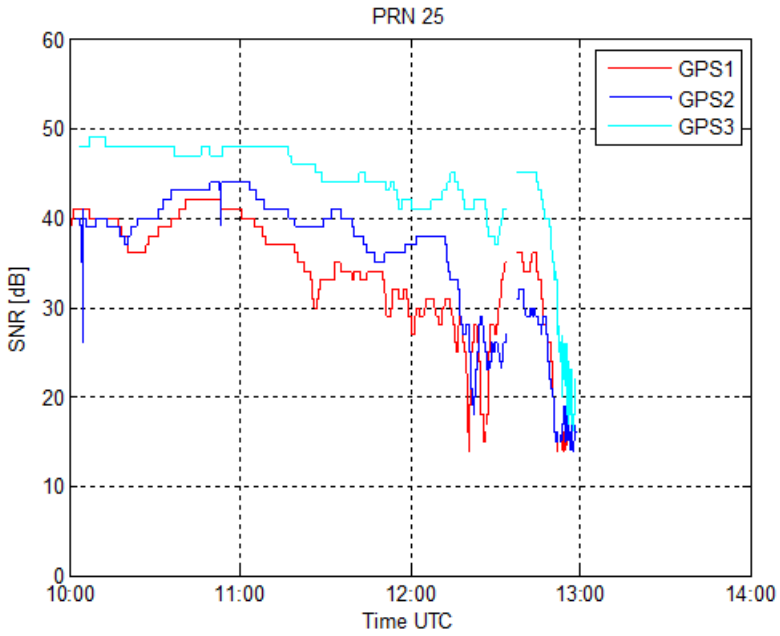

Figure 4.95: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 25 (position 2)

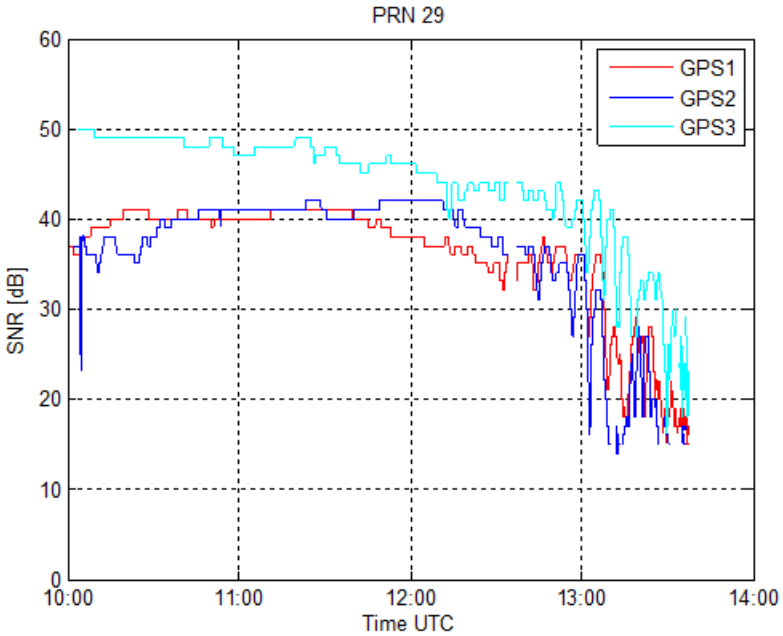

Figure 4.96: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 29 (position 2)

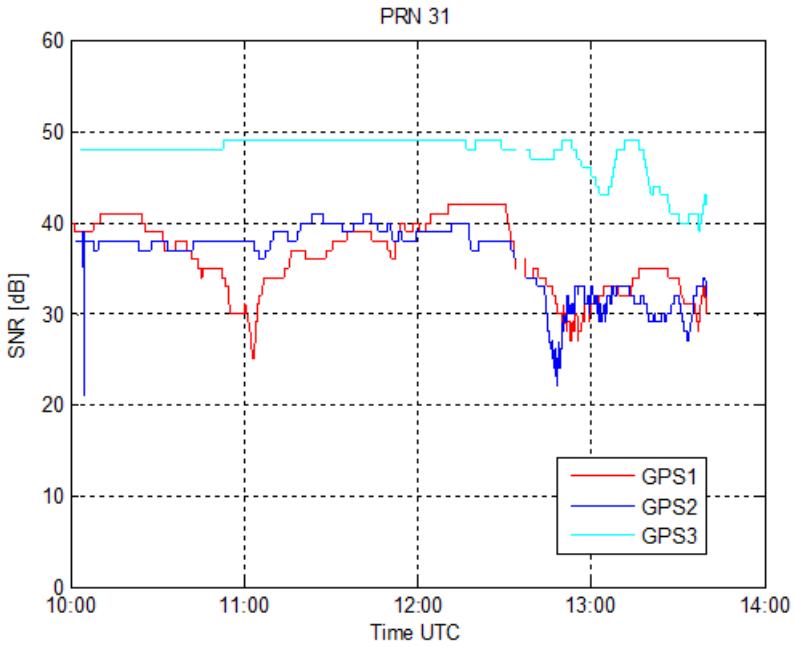

Figure 4.97: SNR (dB) with respect to time (UTC) for the 3 GPS modules for PRN 31 (position 2)

The reported graphs for the SNR obtained for each GPS modules show the effect of rice: the resulting loss for all the satellites in view is around 15 dB for position 1 (under 3 rice sacks), with peaks of 20 dB as reported in 4.92. The data recorded by GPS1 and GPS2 are in good agreement between them. The average difference between the signal level recorded by GPS3 and the other receivers while in position 2 (under 2 rice sacks) is around 10 dB.

The following graphs represent the SNR variation of the different in view satellites recorded by GPS1 and GPS2 in position 1 and position 2, with respect to the data recorded by GPS3. The data referred to position 3 are missing, as in that interval of the time the satellites in view were different, making difficult the comparison. The reported graphs include the satellites in view for the longest period and with the best performance in terms of SNR: PRN 25, 29 and 31.

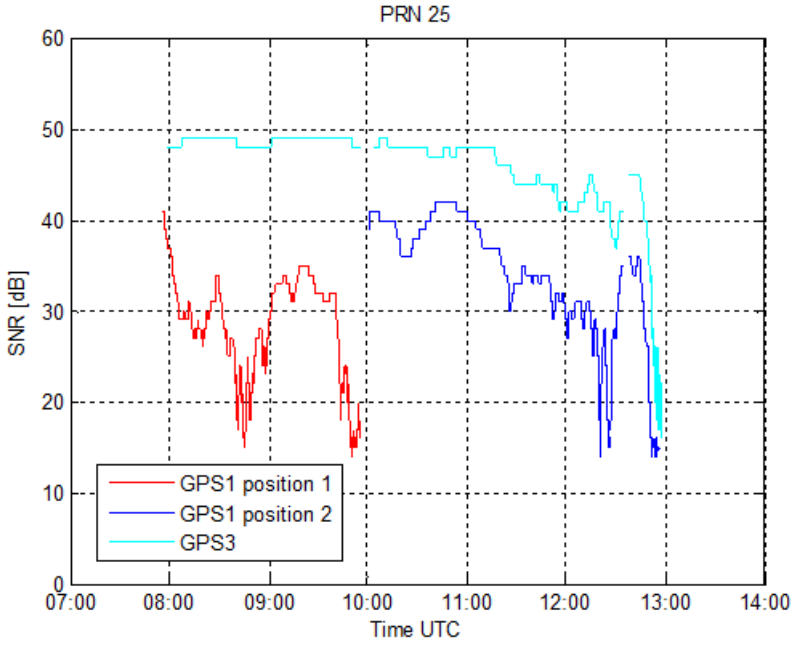

Figure 4.98: SNR (dB) with respect to time (UTC) for GPS3 and GPS1 (position1 and position 2) for PRN 25

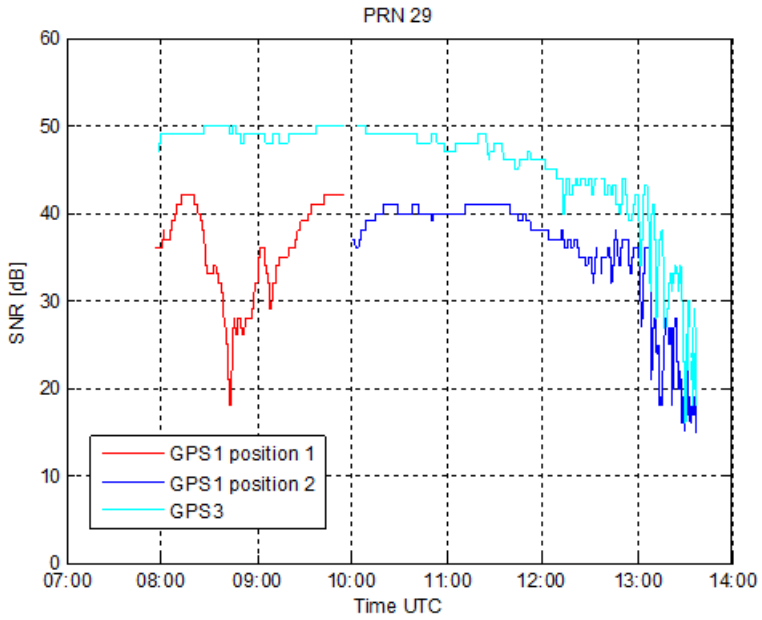

Figure 4.99: SNR (dB) with respect to time (UTC) for GPS3 and GPS1 (position1 and position 2) for PRN 29

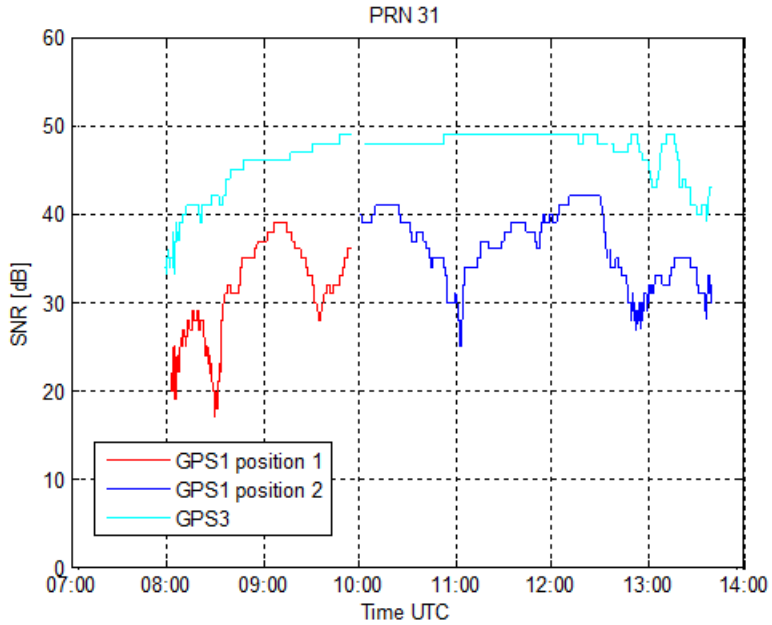

Figure 4.100: SNR (dB) with respect to time (UTC) for GPS3 and GPS1 (position1 and position 2) for PRN 31

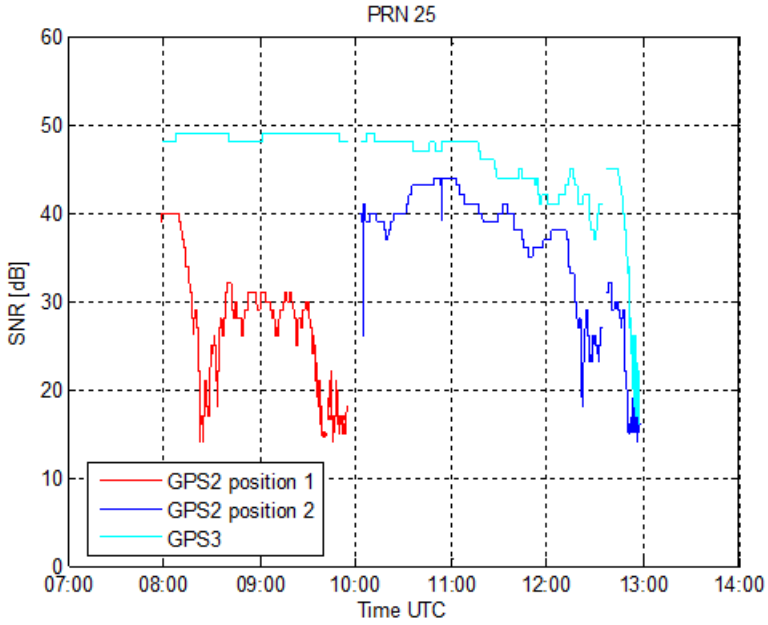

Figure 4.101: SNR (dB) with respect to time (UTC) for GPS3 and GPS2 (position1 and position 2) for PRN 25

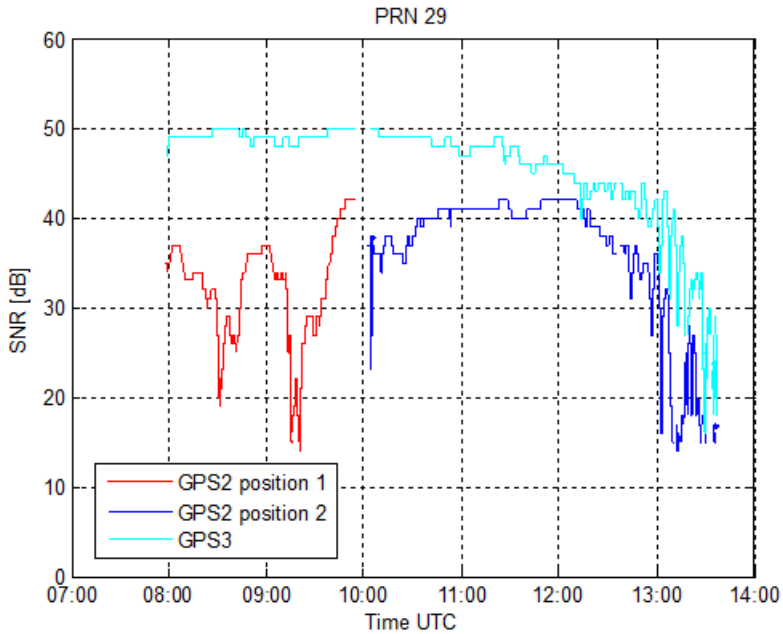

Figure 4.102: SNR (dB) with respect to time (UTC) for GPS3 and GPS2 (position1 and position 2) for PRN 29

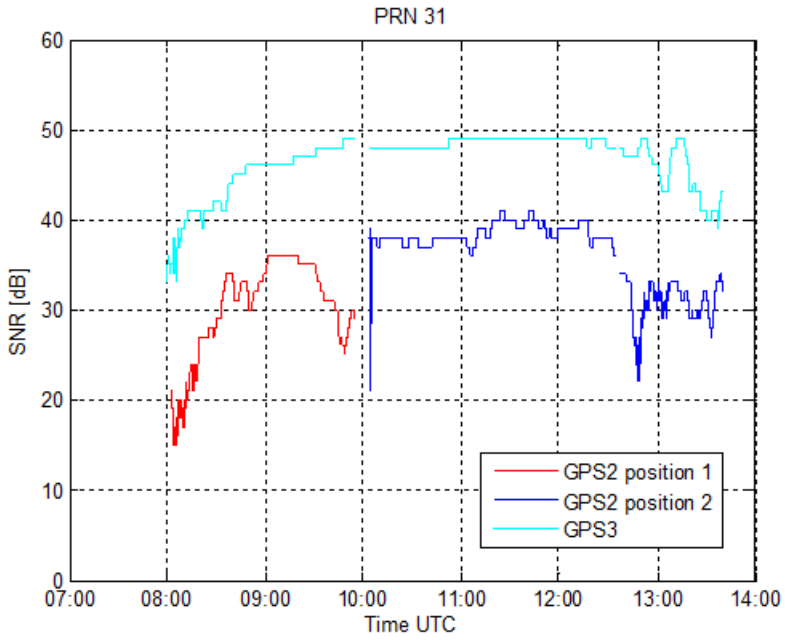

Figure 4.103: SNR (dB) with respect to time (UTC) for GPS3 and GPS2 (position1 and position 2) for PRN 31

The reported graphs show a clear difference between the signal received in position 1 and the signal received in position 2. The average difference is equal to 5 dB, with higher peaks as in 4.98 and in 4.101. Some inversion in the performance has been registered in correspondence of the transition between one position to the other. Moreover it should be kept into account the fact that a fair comparison between the received signal level have to be done in correspondence of the flat part of the signal curve.

The following graphs represent the Horizontal Dilution of Precision (HDOP), the Position Dilution of Precision (PDOP) and the Vertical Dilution of Precision (VDOP).

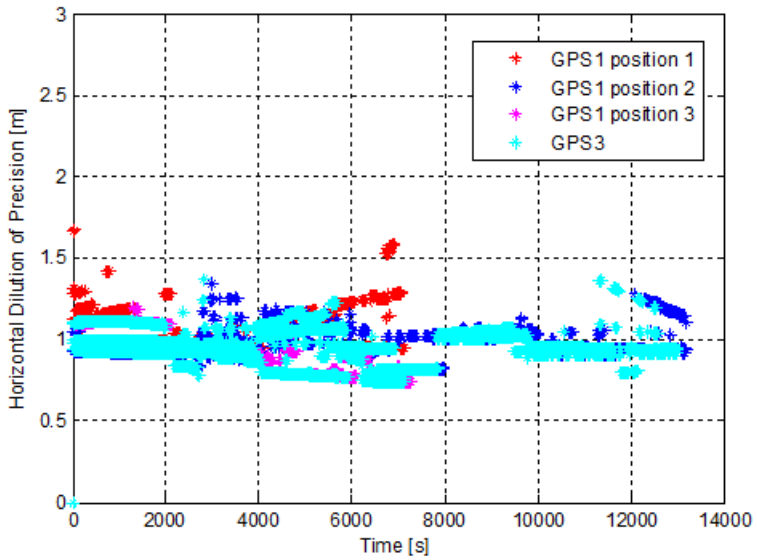

Figure 4.104: Horizontal Dilution of Precision (HDOP) for GPS3 and GPS1 (position1, position2 and position3)

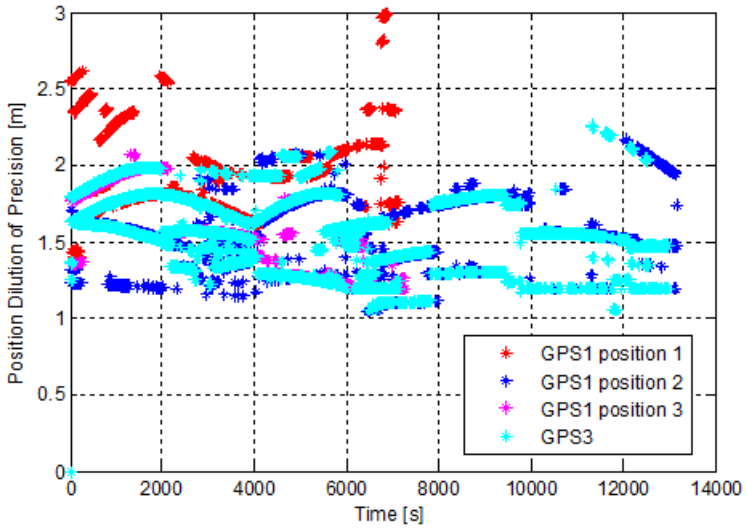

Figure 4.105: Position Dilution of Precision (HDOP) for GPS3 and GPS1 (position1, position2 and position3)

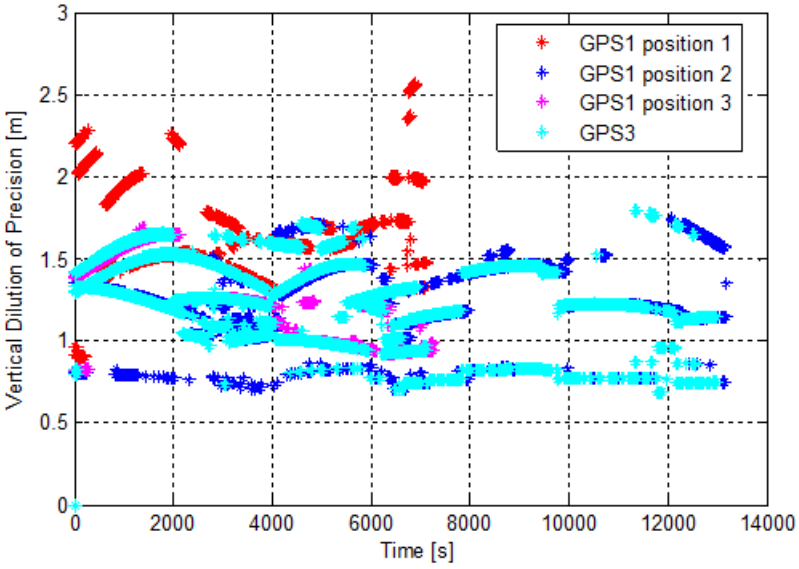

Figure 4.106: Vertical Dilution of Precision (HDOP) for GPS3 and GPS1 (position1, position2 and position3)

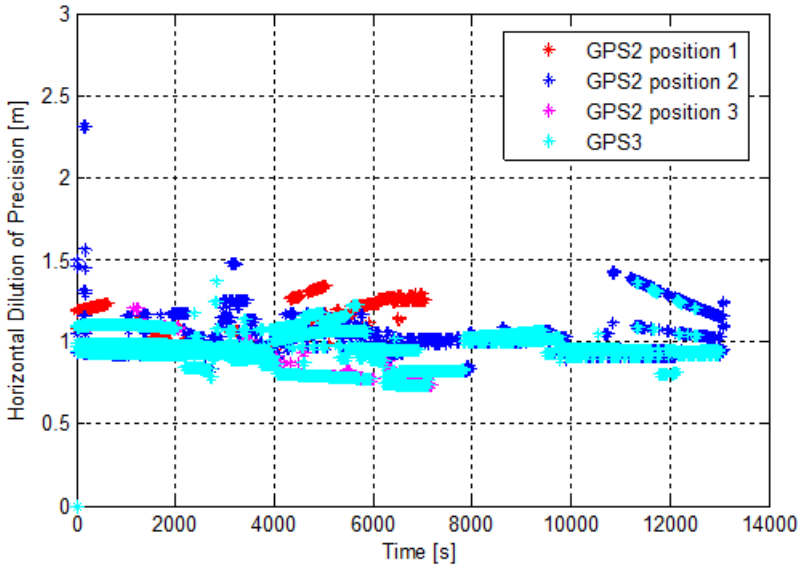

Figure 4.107: Horizontal Dilution of Precision (HDOP) for GPS3 and GPS2 (position1, position2 and position3)

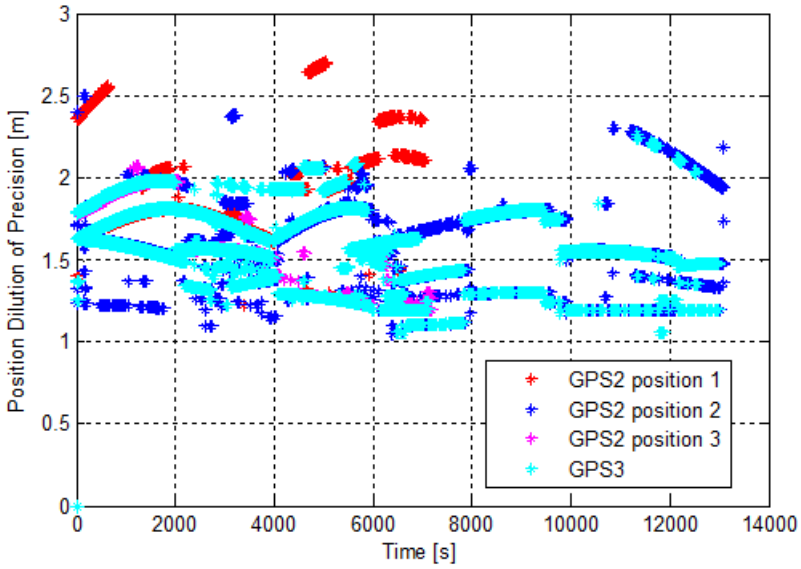

Figure 4.108: Position Dilution of Precision (HDOP) for GPS3 and GPS2 (position1, position2 and position3)

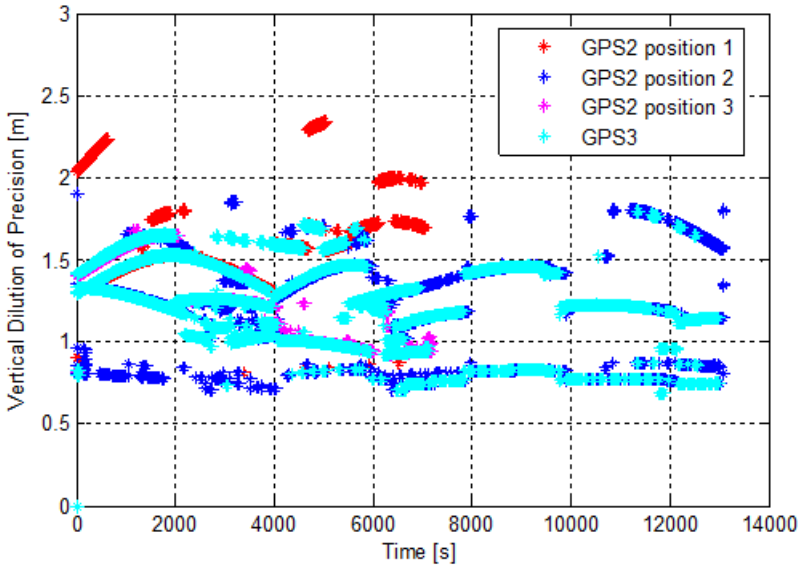

Figure 4.109: Vertical Dilution of Precision (HDOP) for GPS3 and GPS2 (position1, position2 and position3)

The resulting signal quality is very good for the three GPS modules, with values in the range between 1 and 2.5, with just some few peaks. The GPS3 shows, as expected the lowest DOP values. The worst performance are instead obtained with the receivers in position 1 (under the 3 rice sacks), as expected. However the best performance in terms of signal quality is obtained for all the three receivers in the horizontal direction, as shown in 4.104 and 4.107.

The obtained results show clearly the effect of the rice on the received GPS signal and demonstrate that the high sensitivity GPS modules used for the experiment have a sensibility to the variation of the surrounding dielectric material, even if its dielectric characteristics are less strong than the water one. Moreover it appears that to each rice layer corresponds a reduction of the received signal around 5 dB.

4.2.5 Rice - Experiment II b

The last experiment has been performed using a configuration analogous to that used in the previous described experiment, but using two additional GPS receivers equipped with two different antennas. The configuration used is reported in the following figure.

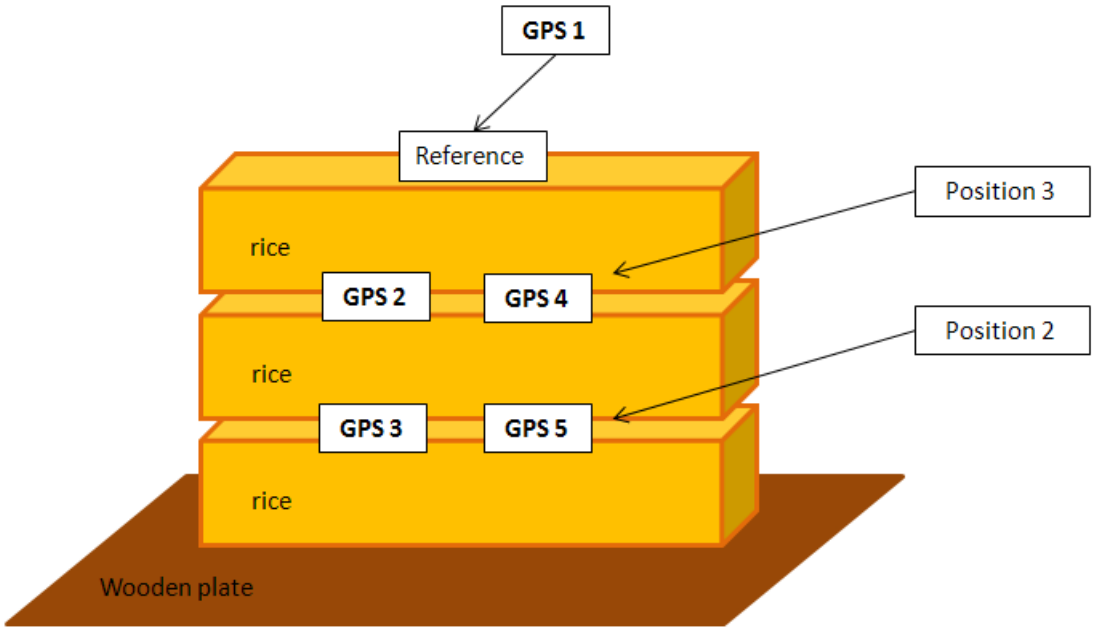

Figure 4.110: View of the used measurement configuration.

The performed measurements confirmed the results obtained with the previous experiment. Moreover the behavior of the new antennas is coherent with respect to the one of the "'old"' antennas. In the following graphs are reported the results obtained in terms of SNR for satellite with PRN 14 for both kind of antennas.

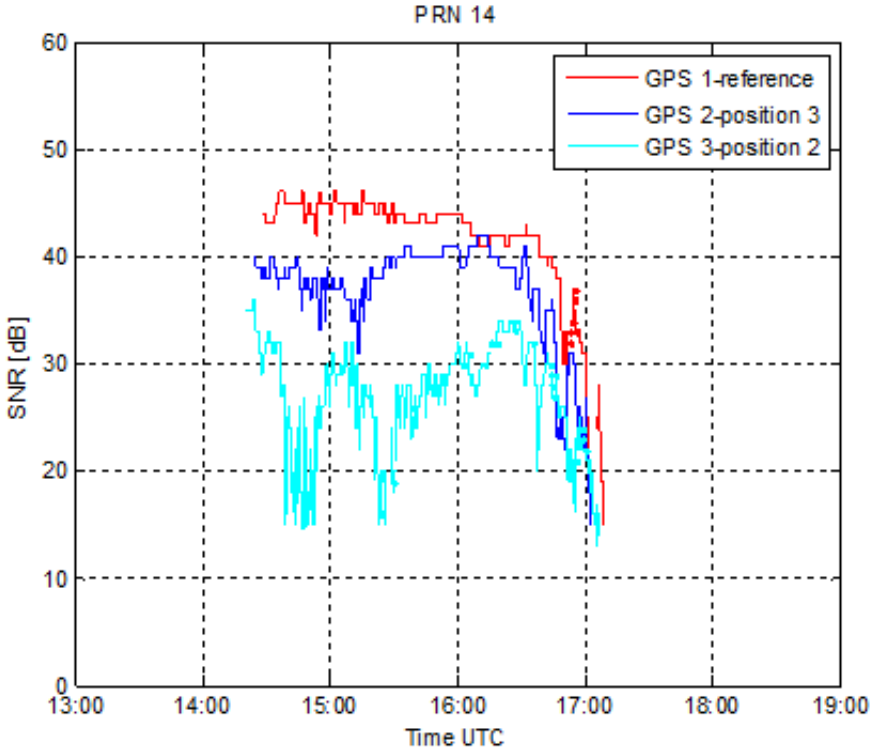

Figure 4.111: SNR (dB) with respect to time (UTC) for GPS3 and GPS2 for PRN 14

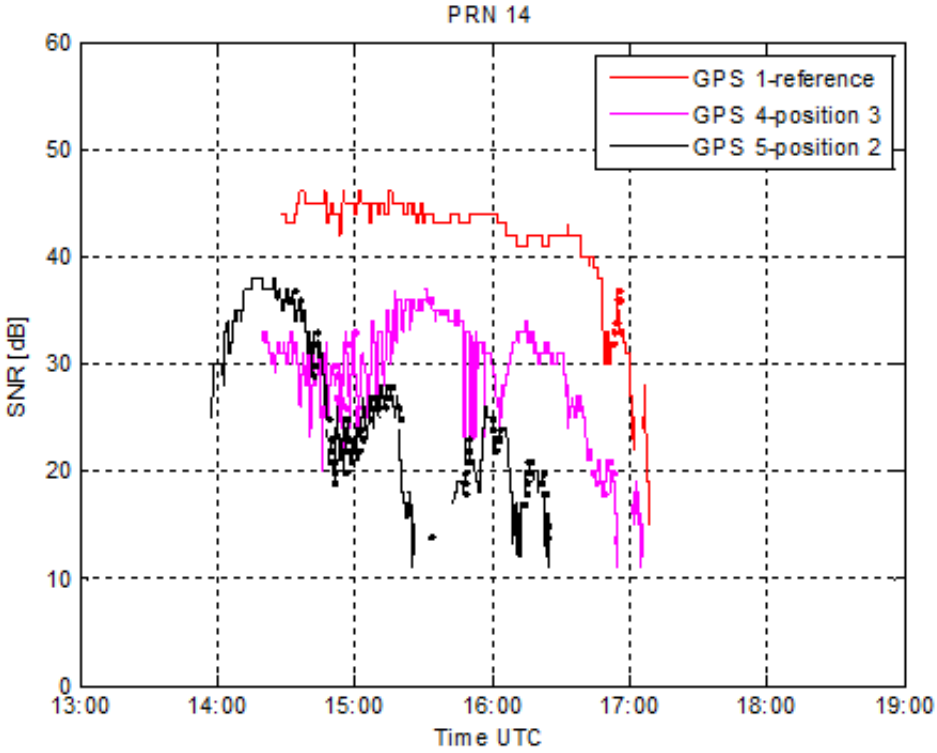

Figure 4.112: SNR (dB) with respect to time (UTC) for GPS4 and GPS5 for PRN 14

The presence of some data inversion has to be further analyzed, performing experiments in a real snow environment in order to exclude any relation with the limited physical dimension of the used setup and the presence of reflections due obstacles.

4.3 Conclusion

The experimental activity and the theoretical analysis of the proposed system showed encouraging results in the perspective of identifying different snow levels and computing the Snow Water Equivalent. The system in fact has been able of distinguish between different dielectrics and also between different depths of the same dielectric. However a further bunch of experiments in real environment will be necessary in order to confirm the expected behavior.

Conclusions

The whole PhD activity has been devoted to the study of the propagation in non-standard conditions. The first activity has been concentrated on the study of the propagation in the proximity of the antenna for the evaluation of the human exposure to electromagnetic fields generated by cellular base stations. The developed study led to the definition of a fast and reliable method that allows the assessment of the field generated by a generic antenna in the near field region of the antenna. The method has been validated by means of comparison with full-wave simulations and measurements data.

The second part of the PhD has been dedicated to the development of sensors for the monitoring of snow conditions. In particular, the second activity concerns the design of a sensor system able to determine the amount of liquid water content into the snow. Such sensor is based on the use of electric and magnetic radiators and it has been development by means of CAD simulations on HFSS. The results of the performed simulations are encouraging and they must be validated by the realization of a prototype to be tested in real environment.

Finally, the last activity has been concentrated on the feasibility study of a snow monitoring system based on commercially available GPS devices. Such system has been tested performing a huge set of experimental activities in controlled environment that led to satisfactory results to be confirmed by means of experiments in real snow.

Appendix A

Near field code

A.1 mainCMD.m

```
function mainCMD


%MAINCMD  main program which implements the method
    described in the
%article "A Fraunhofer based approach for the assessment
     of the field
%radiated in the Fresnel Region of an antenna" (AWPL).
%
%MAINCMD let you set some parameters such as
%
%  - type of antenna
%  - radiated power
%  - radius
%  - sine exponent
%  - type of plot
%  - ....(other features)
%
%
%authors: Roberto Vallauri
%                   Alessandra Carta


clc;
%close all;

param = struct();
```

```
27
  fprintf('\n
      ------------------------------------------------------------------\
      n');
29 fprintf('---------------------- MAIN PROGRAM
      ----------------------------\n');

31 s = strcat('\n
      ------------------------------------------------------------------\
      n',...
    '---------------------- MAIN PROGRAM
        ----------------------------\n ');
33
  fprintf('Select the input:\n');
35 fprintf('1 - Default\n');
  fprintf('2 - Menu\n');
37 in=1;
  % fl = input('Select:    ');
39 %
  %  switch fl
41 %
  %      case 1
43 %          in = 1;
  %
45 %      case 2
  %          in = 2;
47 %
  %  end
49
   if(in == 1)
51    param.AntennaType = '730691';
      param.Cut = 'Vertical';
53    param.FeedingType = 'tsc';
      param.current = 'const';        %'const'--> constant
           current;'sin'  --> sinusoidal current
55    param.shield = 1;               %0 = without shield;
           1 = with shield
      param.delta2_flag = 'sin_step'; %'sin' --> delta'' =
           sin^2; 'sin_step'--> delta'' = sin^2*f_step^2
57                                    %'tot' --> delta'' =
                                           sin^2*f_step^2*
                                          AF^2
```

```
param.fstep_type = 'e2t';        %'e2t'; 'e3t'
if isequal(param.FeedingType,'uni')
    param.Tilt = 0;
end
param.M2s = 0;
if isequal(param.FeedingType,'tsc')
    param.M2s = 40;
end
param.FeedingPower = 1;
param.Radius = 4;
param.SineExponent = 2;
if isequal(param.FeedingType,'tsc')
    param.shift = [0.5 0 0 0.5 1.5 1 1 1.5 1 1 1 1];
    %param.shift=0;
end
if isequal(param.FeedingType,'uni')
    param.shift = 0;
end
param.PlotType = 'all';

if isequal(param.FeedingType,'uni')
    tilt_str = num2str(param.Tilt);
    param.rp_filename = strcat('Uniform/rp_',
        tilt_str,'uni.out');
    param.ne_filename = strcat('Uniform/ne_',
        tilt_str,'uni.out');
end

if isequal(param.FeedingType,'bin')
    param.rp_filename = 'Binomial/rp_bin.out';
    param.ne_filename = 'Binomial/ne_bin.out';
end

if isequal(param.FeedingType,'tsc')
    m2s_str = num2str(param.M2s);
    param.rp_filename = strcat('Tschebyscheff/rp_',
        m2s_str,'tsc.out');
    param.ne_filename = strcat('Tschebyscheff/ne_',
        m2s_str,'tsc.out');
end

s = [s '   ANTENNA TYPE :              Kathrein '
    param.AntennaType '\n'];
```

```
        fprintf(s);
97      s = [s '   CUT :                          ' param.Cut '
            \n'];
        fprintf(s);
99      if isequal(param.FeedingType,'uni');
            temp = 'Uniform';
101     elseif isequal(param.FeedingType,'bin');
            temp = 'Binomial';
103     else isequal(param.FeedingType,'tsc');
            temp = 'Dolph - Tschebyscheff';
105     end
        s = [s '   FEEDING TYPE :                ' temp '\n'];
107     fprintf(s);
        if isequal(param.FeedingType,'uni')
109         temp = num2str(param.Tilt);
            s = [s '   TILT :                          ' temp '
                [deg]  \n'];
111         fprintf(s);
        end
113     temp = num2str(param.M2s);
        s = [s '   MAJOR-to-SIDE LOBE RATIO :  ' temp ' dB
            \n'];
115     s = [s '   TILT :                        ' '0' ' [deg]
              \n'];
        fprintf(s);
117     temp = num2str(param.Radius);
        s = [s '   RADIUS :                      ' temp ' [m]
             \n'];
119     fprintf(s);
        temp = num2str(param.SineExponent);
121     s = [s '   SINE EXPONENT :               ' temp '   \n
            '];
        fprintf(s);
123     temp = num2str(param.shift);
        s = [s '   SHIFT :                       ' temp '   \n
            '];
125     fprintf(s);

127 end


129
  if(in == 2)
131     %Selezione del modello di antenna
```

```
fprintf('\n\n\n ------> ANTENNA TYPE    \n');
fprintf(' 1 - Kathrein 730691\n');
fprintf(' 2 - Kathrein 742215 (not available yet)  \
    n');

fl = input('Select:    ');

switch fl

    case 1
        param.AntennaType = '730691';

    case 2
        param.AntennaType = '742215';

end

clc;
s = [s '   ANTENNA TYPE :              Kathrein '
    param.AntennaType '\n'];
fprintf(s);

%Selezione taglio (verticale o orizzontale)
fprintf('\n\n\n ------> CUT   \n');
fprintf(' 1 - Vertical\n');
fprintf(' 2 - Horizontal (not available yet)  \n');

fl = input('Select:    ');

switch fl

    case 1
        param.Cut = 'Vertical';

    case 2
        param.Cut = 'Horizontal';

end
clc;
s = [s '   CUT :                          ' param.Cut '
    \n'];
fprintf(s);
```

```
%Selezione tipo di alimentazione
fprintf('\n\n\n ------> FEEDING TYPE   \n');
fprintf(' 1 - Uniform\n');
fprintf(' 2 - Binomial \n');
fprintf(' 3 - Dolph - Tschebyscheff\n');

fl = input('Select:    ');

switch fl

    case 1
        param.FeedingType = 'uni';

    case 2
        param.FeedingType = 'bin';

    case 3
        param.FeedingType = 'tsc';

end
clc;
if isequal(param.FeedingType,'uni');
    temp = 'Uniform';
elseif isequal(param.FeedingType,'bin');
    temp = 'Binomial';
else isequal(param.FeedingType,'tsc');
    temp = 'Dolph - Tschebyscheff';
end

s = [s '   FEEDING TYPE :              ' temp '\n'];
fprintf(s);

%Selezione dell angolo di tilt (se alimentazione
    uniforme)
if isequal(param.FeedingType,'uni')

    fprintf('\n ------> TILT   \n');
    fprintf(' 1 - 0 deg  \n');
    fprintf(' 2 - 2 deg  \n');
    fprintf(' 3 - 5 deg  \n');
    fprintf(' 4 - 10 deg  \n');

    fl = input('Select:    ');
```

```
215         switch fl

217             case 1
                    param.Tilt = 0;
219
                case 2
221                 param.Tilt = 2;

223             case 3
                    param.Tilt = 5;
225
                case 4
227                 param.Tilt = 10;

229         end
            clc;
231         temp = num2str(param.Tilt);
            s = [s '   TILT :                          ' temp '
                [deg]  \n'];
233         fprintf(s);
    end
235
    %Selezione del major-to-side lobe ratio (se
        alimentazione Tschebyscheff)
237 if isequal(param.FeedingType,'tsc')

239         fprintf('\n ------> MAJOR-to-SIDE LOBE RATIO   \
                n');
            fprintf(' 1 - 15 dB  \n');
241         fprintf(' 2 - 20 dB  \n');
            fprintf(' 3 - 25 dB  \n');
243         fprintf(' 4 - 30 dB  \n');
            fprintf(' 5 - 35 dB  \n');
245         fprintf(' 6 - 40 dB  \n');

247         fl = input('Select:    ');

249         switch fl

251             case 1
                    param.M2s = 15;
253
```

```
            case 2
255             param.M2s = 20;

257         case 3
                param.M2s = 25;
259
            case 4
261             param.M2s = 30;

263         case 5
                param.M2s = 35;
265
            case 6
267             param.M2s = 40;

269     end
        clc;
271     temp = num2str(param.M2s);
        s = [s '   MAJOR-to-SIDE LOBE RATIO :  ' temp '
            dB  \n'];
273     s = [s '   TILT :                      ' '0' ' [
            deg]  \n'];
        fprintf(s);
275 end

277 %Posso a questo punto con i dati a disposizione
        determinare il nome dei
    %file NEC-2 di output, ovvero Radiation Pattern (far
         field) e Near Electric
279 %Field. Ricordo qual'e il formato :
    %
281 %Uniforme: rp_5uni (radiation pattern,5 gradi di
        tilt,alimentazione
    %                   uniforme)
283 %
    %Tschebyscheff: ne_20tsc (near electric field, 20 dB
         di major-to-side lobe
285 %                           ratio,alimentazione Dolph
        - Tschebyscheff)

287 if isequal(param.FeedingType,'uni')
        tilt_str = num2str(param.Tilt);
289     param.rp_filename = strcat('Uniform/rp_',
```

```
            tilt_str,'uni.out');
        param.ne_filename = strcat('Uniform/ne_',
            tilt_str,'uni.out');
    end

    if isequal(param.FeedingType,'bin')
        param.rp_filename = 'Binomial/rp_bin.out';
        param.ne_filename = 'Binomial/ne_bin.out';
    end

    if isequal(param.FeedingType,'tsc')
        m2s_str = num2str(param.M2s);
        param.rp_filename = strcat('Tschebyscheff/rp_',
            m2s_str,'tsc.out');
        param.ne_filename = strcat('Tschebyscheff/ne_',
            m2s_str,'tsc.out');
    end

    %Selezione potenza con cui vado ad alimentare l'
        antenna
    fprintf('\n ------> FEEDING POWER   \n');
    fprintf(' 1 - 1 W             \n');
    fprintf(' 2 - (manual)        \n');


    fl = input('Select:    ');

        switch fl

            case 1
                param.FeedingPower = 1;

            case 2
                param.FeedingPower =input('Insert
                    Feeding Power [W] = ');

        end
    clc;
    temp = num2str(param.FeedingPower);
    s = [s '   FEEDING POWER :             ' temp ' [W]
         \n'];
    fprintf(s);
```

```
%Selezione distanza alla quale voglio calcolare il
    campo
fprintf('\n ------> RADIUS   \n');
fprintf(' 1 -  4  m            \n');
fprintf(' 2 -  6  m            \n');
fprintf(' 3 -  8  m            \n');
fprintf(' 4 -  10 m            \n');
fprintf(' 5 -  12 m           \n');
fprintf(' 6 -  (manual)       \n');


fl = input('Select:    ');

    switch fl

        case 1
            param.Radius = 4;

        case 2
            param.Radius = 6;

        case 3
            param.Radius = 8;

        case 4
            param.Radius = 10;

        case 5
            param.Radius = 12;

        case 6
            param.Radius = input('Insert Radius [m]
                = ');

    end
clc;
temp = num2str(param.Radius);
s = [s '   RADIUS :                    ' temp ' [m]
     \n'];
fprintf(s);

%Selezione esponente a cui elevare il seno
fprintf('\n ------> SINE EXPONENT   \n');
```

```
        fprintf(' 1 -  2              \n');
367     fprintf(' 2 -  3              \n');
        fprintf(' 3 -  4              \n');
369     fprintf(' 4 -  (manual)         \n');

371     fl = input('Select:    ');

373     switch fl

375         case 1
                param.SineExponent = 2;
377
            case 2
379             param.SineExponent = 3;

381         case 3
                param.SineExponent = 4;
383
            case 4
385             param.SineExponent = input('Insert Sine
                    Exponent = ');

387     end
        clc;
389     temp = num2str(param.SineExponent);
        s = [s '   SINE EXPONENT :              ' temp '   \n
            '];
391     fprintf(s);

393     param.shift = input('\nInsert shift = ');
        clc;
395     temp = num2str(param.shift);
        s = [s '   SHIFT :                      ' temp '   \n
            '];
397     fprintf(s);

399     %Selezione tipo di plottaggio che si vuole
            visualizzare
        fprintf('\n ------> PLOT TYPE   \n');
401     fprintf(' 1 -  Cut (Fig. 2 AWPL)   \n');
        fprintf(' 2 -  Additive Incremental Terms (Fig. 4
            AWPL) \n');
403     fprintf(' 3 -  Integrals (Fig. 3 AWPL)  \n');
```

```
        fprintf(' 4 -  All Plots    \n');
405
        fl = input('Select:    ');
407
            switch fl
409
                case 1
                    param.PlotType = 'cut';
411
                case 2
                    param.PlotType = 'ait';
413
                case 3
                    param.PlotType = 'int';
415
                case 4
                    param.PlotType = 'all';
417
            end
419     clc;
        s = [s '\n\n\n'];
421     fprintf(s);
    end
423
    %Svolge le operazioni e plotta quanto richiesto (anche
        su file .tif)
425 param.handle_figure = plotFields(param.Radius,param.
        FeedingPower,...
        param.rp_filename,param.ne_filename,param.
            SineExponent,...
427     param.PlotType,param.shift,param.M2s,param.current,
            param.shield,param.delta2_flag,param.fstep_type
            );

429 %  fprintf('\nDo you want to plot the figure on a .tif
        file?\n');
    %  fprintf(' 1 -  NO  \n');
431 %  fprintf(' 2 -  YES \n');
    %  fl = input('Select:    ');
433 %
    %  if fl == 2
435 %
    %  s = input('\n\nInsert name of the output file (
```

```
      without .tif):','s');
%
%
% file_name_out = sprintf('Output_files/%s.tif',s);
%
% print(param.handle_figure, '-dtiff', file_name_out);
%
%
%  else return,
%  end;
```

A.2 EfieldLinearCurrent.m

```
% Computes the complex amplitude of electric field radiated by a linear
% constant current according to Fresnel and Fraunhofer approximation.
%
% Current distribution:
%       Je=I0*exp(-i (2*pi/lambda)*cos(thetaM)*z )
%           -D/2<= z <= D/2
%
%
%                              /|\ axis z
%                               |
%                               +  D/2
%                               |
%                               |
%                               +--->
%                               |
%                               |
%                               + -D/2
%                               |
%
% I0 ................. current [A]
% theta, r ........... angle [deg] with respect to z positive axis, and
%                      distance [m] defining position at wich the
%                      radiated field will be computed
% D .................. current length [m]
% lambda  ............ vawelength [m];
% thetaM  ............ position (theta [deg]) of pattern peak;
%
% Efs ................ Fresnel approximation of electric field [V/m] at
%                      distance r and angular position theta
% Efr ................ Fraunhofer  approximation of electric field [V/m] at
%                      distance r and angular position theta
% DE2=(|Efs|^2-|Efr|^2)/Z0
```

```
%
% output: A,B,C correspond to DE2, Efs, Efr as specified by the input
% variables first, second, third
%
% The electric fieed in Fresnel region (Efs) is based on Fresnel (C, S)
% integrals. The computation of such a integrals is made exactly (by means
% of function Fresenl.m) or by an approximated formula (function
% FresnelApprox.m).
% The input variable FresnelComp select the Fresnel integral computation
% according to:
% FresnelComp='exact'  --> exact Fresnel integral computation
% FresnelComp='approx' --> approximated Fresnel integral computation


function [A,B,C, f_step] = EfieldLinearCurrent(FresnelComp,I0,theta,r,...
D,ff_filename,lambda,current_type,shield,fstep_type,thetaM,first,second,third)

c0 = 299.792458;      % free space light speed  [mm/ns]
Z0 = 4*pi*c0*10^-1;  % free space impedance [ohm] (approx. Z0=120*pi)

ui   = complex(0,1); % imaginary unit
d2r = pi/180;        % deg to radian

method_sel = 2;       % 1 = element length fixed = lambda/2;
                      % 2 = elements length variable and related to N
N=6;
% index = 2; %1:N=4; 2:N=6; 3:N=8; 4:N=10; 5:N=12; 6:N=14; 7:N=16.
% N=pair(index,1);

```

```
X=0.9;
if method_sel == 1
    %method 1
    elem_length=lambda/2;
    d_lambda=((D/lambda)-X)/(N-1);
    d=d_lambda*lambda;
    pos_z = linspace(0,d*(N-1),N)-d*(N-1)/2;
else
    %method 2
    pos_z = linspace(-D/2,D/2,N+1);
    elem_length=abs(pos_z(1)-pos_z(2));
    d=abs((pos_z(1)+pos_z(2))/2-(pos_z(2)+pos_z(3))/2);
    d_lambda=d/lambda;
end
np=normalization_pirr_uni(I0,theta,r,D,lambda,thetaM,N,d
    ,method_sel,...
elem_length,FresnelComp,ff_filename,current_type,shield)
    ;
cf = -ui*Z0/(2*r*lambda)*I0*exp(-ui*2*pi*r/lambda)*np; %
     Common factor

t   =  cos(d2r*theta);
csi=  cos(d2r*thetaM);


sth=  sin(d2r*theta);


ntheta=length(theta);

if shield == 0 %non viene simulata la presenza dello
    schermo
    x0=0;       %distance between the antenna and the
        shield
    beta=x0/r;
else
    N=2*N;
    x0=0.054;   %distance between the antenna and the
        shield
    beta=x0/r;
    sigma_ant=beta^2-(2*beta*sqrt(1-t.^2));
    sigma_image=beta^2+(2*beta*sqrt(1-t.^2));
    cf_sigma_ant=exp(-ui*2*pi*r*((1/2)*sigma_ant-(1/8)*
```

```
        sigma_ant.^2)/lambda);
    %(-1)-->phase shift of 180 deg for the image (exp(j*
        pi))
    cf_sigma_image=(-1)*exp(-ui*2*pi*r*((1/2)*
        sigma_image-(1/8)*sigma_image.^2)/lambda);
end

if method_sel == 1
    %element length = lambda/2
    for j=1:length(pos_z)
        z1(j)=pos_z(j)-(elem_length/2);
        z2(j)=pos_z(j)+(elem_length/2);
        zm(j)=0.5*(z1(j)+z2(j));
        zd(j)=0.5*(z2(j)-z1(j));
    end
else
    %element length variable
    for j=1:length(pos_z)-1
        z1(j)=pos_z(j);
        z2(j)=pos_z(j+1);
        zm(j)=0.5*(z1(j)+z2(j));
        zd(j)=0.5*(z2(j)-z1(j));
    end
end
%z1=zm-zd
%z2=zm+zd

x=1;
y=1;
for i=1:2*length(z1)
    if mod(i,2) == 0
        z(i)=z2(x);
        x=x+1;
    else
        z(i)=z1(y);
        y=y+1;
    end
end


% preallocation to increase speed
Efr=complex(10^-100,10^-100)*zeros(1,ntheta);
Efr_ant=Efr;
```

```
Efr_image=Efr;
Efr_tot=Efr;
Efs=Efr;
Efs_tot=Efr;
DE2=Efr;

%FRAUNHOFER
if(strcmp(current_type,'const') == 1)
    if(shield == 0) %corrente costante, senza schermo
        for j=1:N
            rp=exp((ui*zm(j)*(t-csi)*2*pi)./lambda);
            Efr=2*zd(j)*cf.*rp.*sth.*sinc(2*zd(j)*(t-csi
                )/lambda);
            Efr_tot = Efr_tot+Efr;
        end
    Efr=Efr_tot;
    else  %corrente costante + schermo
        for j=1:N
            if j <= N/2
                rp_ant=cf_sigma_ant.*exp((ui*zm(j)*(t
                    .*(1-0.5*sigma_ant)-csi)*2*pi)./
                    lambda);
                Efr=zd(j)*cf.*sth.*(rp_ant.*sinc(2*zd(j)
                    *(t.*(1-0.5*sigma_ant)-csi)/lambda))
                    ;
                Efr_tot = Efr_tot+Efr;
            else
                rp_image=cf_sigma_image.*exp((ui*zm(j-(N
                    /2))*(t.*(1-0.5*sigma_image)-csi)*2*
                    pi)./lambda);
                Efr=zd(j-(N/2))*cf.*sth.*(rp_image.*sinc
                    (2*zd(j-(N/2))*(t.*(1-0.5*
                    sigma_image)-csi)/lambda));
                Efr_tot = Efr_tot+Efr;
            end
        end
    Efr=Efr_tot;
    end
else
    if(shield == 0) %corrente sinusoidale, senza schermo
        for j=1:N
            rp1=exp((ui*zm(j)*(t-csi+1)*2*pi)./lambda).*
                exp(-ui*zm(j)*2*pi/lambda);
```

```
            rp2=exp((ui*zm(j)*(t-csi-1)*2*pi)./lambda).*
                exp(ui*zm(j)*2*pi/lambda);
173         Efr=zd(j)*cf.*sth.*((rp1.*sinc(2*zd(j)*(t-
                csi+1)/lambda))+(rp2.*sinc(2*zd(j)*(t-
                csi-1)/lambda)));
            Efr_tot = Efr_tot+Efr;
175     end
        Efr=Efr_tot;
177 else   %corrente sinusoidale + schermo
        for j=1:N
179         if j <= N/2
                rp1_ant=cf_sigma_ant.*exp((ui*zm(j)*(t
                    .*(1-0.5*sigma_ant)-csi+1)*2*pi)./
                    lambda).*exp(-ui*zm(j)*2*pi/lambda);
181             rp2_ant=cf_sigma_ant.*exp((ui*zm(j)*(t
                    .*(1-0.5*sigma_ant)-csi-1)*2*pi)./
                    lambda).*exp(ui*zm(j)*2*pi/lambda);
                Efr=zd(j)*cf.*sth.*((rp1_ant.*sinc(2*zd(
                    j)*(t.*(1-0.5*sigma_ant)-csi+1)/
                    lambda))+(rp2_ant.*sinc(2*zd(j)*(t
                    .*(1-0.5*sigma_ant)-csi-1)/lambda)))
                    ;
183             Efr_tot = Efr_tot+Efr;
            else
185             rp1_image=cf_sigma_image.*exp((ui*zm(j-(
                    N/2))*(t.*(1-0.5*sigma_image)-csi+1)
                    *2*pi)./lambda).*exp(-ui*zm(j-(N/2))
                    *2*pi/lambda);
                rp2_image=cf_sigma_image.*exp((ui*zm(j-(
                    N/2))*(t.*(1-0.5*sigma_image)-csi-1)
                    *2*pi)./lambda).*exp(ui*zm(j-(N/2))
                    *2*pi/lambda);
187             Efr=zd(j-(N/2))*cf.*sth.*((rp1_image.*
                    sinc(2*zd(j-(N/2))*(t.*(1-0.5*
                    sigma_image)-csi+1)/lambda))+(
                    rp2_image.*sinc(2*zd(j-(N/2))*(t
                    .*(1-0.5*sigma_image)-csi-1)/lambda)
                    ));
                Efr_tot = Efr_tot+Efr;
189         end
        end
191 Efr=Efr_tot;
    end
```

```
193 end
    %
195 %FRESNEL
    if (strcmp(current_type,'const') == 1) % CONSTANT
        CURRENT
197     if(shield == 0) %constant current without shield
            for j=1:N
199             for k=1:ntheta
                tk=t(k);
201             if abs(tk)~=1
                    % Fresnel integrals
203                 k1c=sqrt(2*r/(lambda*(1-tk^2)));
                    k1=k1c*((1-tk^2)*(z1(j)/(r))-(tk-csi));
205                 k2=k1c*((1-tk^2)*(z2(j)/(r))-(tk-csi));
                    if strcmpi(FresnelComp,'approx')
207                     [Ck1,Sk1] = FresnelApprox(k1);
                        [Ck2,Sk2] = FresnelApprox(k2);
209                 else
                        [Ck1,Sk1] = Fresnel(k1);
211                     [Ck2,Sk2] = Fresnel(k2);
                    end
213                 %Fresnel field
                    Efs(k)= (cf*sqrt(r*lambda/(2*(1-tk^2)))*
                        sth(k)*...
215                     complex(Ck2-Ck1,Sk1-Sk2)*...
                        exp(ui*pi*r/lambda*(-2+(tk-csi)
                            ^2/(1-tk^2)))); % [V/m]
217             else
                    Efs(k)=Efr(k);
219             end
            end
221         Efs_tot = Efs_tot + Efs;
        end
223     Efs=Efs_tot;
        else %constant current with shield
225       for j=1:N
            for k=1:ntheta
227         tk=t(k);
            sigmak_ant=sigma_ant(k);
229         sigmak_image=sigma_image(k);
            if abs(tk)~=1
231             if j <= N/2
                    % Fresnel integrals
```

```
233                 k1c=sqrt(2*r/(lambda*(1-tk^2-0.5*
                        sigmak_ant)));
                    k1=k1c*((1-tk^2-0.5*sigmak_ant)*(z1(
                        j)/(r))-(tk*(1-0.5*sigmak_ant)-
                        csi));
235                 k2=k1c*((1-tk^2-0.5*sigmak_ant)*(z2(
                        j)/(r))-(tk*(1-0.5*sigmak_ant)-
                        csi));
                else
237                 k1c=sqrt(2*r/(lambda*(1-tk^2-0.5*
                        sigmak_image)));
                    k1=k1c*((1-tk^2-0.5*sigmak_image)*(
                        z1(j-(N/2))/(r))-(tk*(1-0.5*
                        sigmak_image)-csi));
239                 k2=k1c*((1-tk^2-0.5*sigmak_image)*(
                        z2(j-(N/2))/(r))-(tk*(1-0.5*
                        sigmak_image)-csi));
                end
241             if strcmpi(FresnelComp,'approx')
                    [Ck1,Sk1] = FresnelApprox(k1);
243                 [Ck2,Sk2] = FresnelApprox(k2);
                else
245                 [Ck1,Sk1] = FresnelApprox(k1);
                    [Ck2,Sk2] = FresnelApprox(k2);
247             end
                %Fresnel field
249             if j <= N/2
                    Efs(k)= (0.5*cf*sqrt(r*lambda/(2*(1-
                        tk^2-0.5*sigmak_ant)))*sth(k))
                        *...
251                 (cf_sigma_ant(k).*(complex(Ck2-Ck1,
                        Sk1-Sk2)*...
                    exp(ui*pi*r/lambda*((tk*(1-0.5*
                        sigmak_ant)-csi)^2/(1-tk^2-0.5*
                        sigmak_ant))))); % [V/m]
253             else
                    Efs(k)= (0.5*cf*sqrt(r*lambda/(2*(1-
                        tk^2-0.5*sigmak_image)))*sth(k))
                        *...
255                 (cf_sigma_image(k).*(complex(Ck2-Ck1
                        ,Sk1-Sk2)*...
                    exp(ui*pi*r/lambda*((tk*(1-0.5*
                        sigmak_image)-csi)^2/(1-tk
```

```
                        ^2-0.5*sigmak_image))))); % [V/m
                        ]
257                 end
            else
259                 Efs(k)=Efr(k);

261             end
        end
263     Efs_tot = Efs_tot + Efs;
        end
265   Efs=Efs_tot;
    end
267 %SINUSOIDAL CURRENT
  else %sinusoidal current without shield
269     if(shield == 0)
            for j=1:N
271                 for k=1:ntheta
                        tk=t(k);
273                     if abs(tk)~=1
                            % Fresnel integrals
275                         k1c=sqrt(2*r/(lambda*(1-tk^2)));

277                         k11=k1c*((1-tk^2)*(z1(j)/(r))-(tk-
                                csi+1));
                            k21=k1c*((1-tk^2)*(z2(j)/(r))-(tk-
                                csi+1));
279                         k12=k1c*((1-tk^2)*(z1(j)/(r))-(tk-
                                csi-1));
                            k22=k1c*((1-tk^2)*(z2(j)/(r))-(tk-
                                csi-1));
281
                            if strcmpi(FresnelComp,'approx')
283                             [Ck11,Sk11] = FresnelApprox(k11)
                                    ;
                                [Ck21,Sk21] = FresnelApprox(k21)
                                    ;
285                             [Ck12,Sk12] = FresnelApprox(k12)
                                    ;
                                [Ck22,Sk22] = FresnelApprox(k22)
                                    ;
287                         else
                                [Ck11,Sk11] = FresnelApprox(k11)
                                    ;
```

```
289                         [Ck21,Sk21] = FresnelApprox(k21)
                                ;
                            [Ck12,Sk12] = FresnelApprox(k12)
                                ;
291                         [Ck22,Sk22] = FresnelApprox(k22)
                                ;
                        end
293                     %Fresnel field
                        Efs(k)= (0.5*cf*sqrt(r*lambda/(2*(1-
                            tk^2)))*sth(k))*...
295                         ((complex(Ck21-Ck11,Sk11-Sk21)
                                *...
                            exp(ui*pi*r/lambda*((tk-csi+1)
                                ^2/(1-tk^2)))*exp(-ui*zm(j)
                                *2*pi/lambda))+...
297                         (complex(Ck22-Ck12,Sk12-Sk22)
                                *...
                            exp(ui*pi*r/lambda*((tk-csi-1)
                                ^2/(1-tk^2))))*exp(ui*zm(j)
                                *2*pi/lambda)); % [V/m]
299                 else
                        Efs(k)=Efr(k);
301                 end
            end
303         Efs_tot = Efs_tot + Efs;
        end
305 Efs=Efs_tot;
    else %sinusoidal current with shield
307   for j=1:N
            for k=1:ntheta
309         tk=t(k);
            sigmak_ant=sigma_ant(k);
311         sigmak_image=sigma_image(k);
            if abs(tk)~=1
313             if j <= N/2
                    % Fresnel integrals
315                 k1c=sqrt(2*r/(lambda*(1-tk^2-0.5*
                        sigmak_ant)));
                    k11=k1c*((1-tk^2-0.5*sigmak_ant)*(z1
                        (j)/(r))-(tk*(1-0.5*sigmak_ant)-
                        csi+1));
317                 k21=k1c*((1-tk^2-0.5*sigmak_ant)*(z2
                        (j)/(r))-(tk*(1-0.5*sigmak_ant)-
```

```
                csi+1));
            k12=k1c*((1-tk^2-0.5*sigmak_ant)*(z1
                (j)/(r))-(tk*(1-0.5*sigmak_ant)-
                csi-1));
            k22=k1c*((1-tk^2-0.5*sigmak_ant)*(z2
                (j)/(r))-(tk*(1-0.5*sigmak_ant)-
                csi-1));
        else
            k1c=sqrt(2*r/(lambda*(1-tk^2-0.5*
                sigmak_image)));
            k11=k1c*((1-tk^2-0.5*sigmak_image)*(
                z1(j-(N/2))/(r))-(tk*(1-0.5*
                sigmak_image)-csi+1));
            k21=k1c*((1-tk^2-0.5*sigmak_image)*(
                z2(j-(N/2))/(r))-(tk*(1-0.5*
                sigmak_image)-csi+1));
            k12=k1c*((1-tk^2-0.5*sigmak_image)*(
                z1(j-(N/2))/(r))-(tk*(1-0.5*
                sigmak_image)-csi-1));
            k22=k1c*((1-tk^2-0.5*sigmak_image)*(
                z2(j-(N/2))/(r))-(tk*(1-0.5*
                sigmak_image)-csi-1));
        end
        if strcmpi(FresnelComp,'approx')
            [Ck11,Sk11] = FresnelApprox(k11);
            [Ck21,Sk21] = FresnelApprox(k21);
            [Ck12,Sk12] = FresnelApprox(k12);
            [Ck22,Sk22] = FresnelApprox(k22);
        else
            [Ck11,Sk11] = FresnelApprox(k11);
            [Ck21,Sk21] = FresnelApprox(k21);
            [Ck12,Sk12] = FresnelApprox(k12);
            [Ck22,Sk22] = FresnelApprox(k22);
        end
        %Fresnel field
        if j <= N/2
            Efs(k)= (0.5*cf*sqrt(r*lambda/(2*(1-
                tk^2-0.5*sigmak_ant)))*sth(k))
                *...
            (cf_sigma_ant(k).*(complex(Ck21-Ck11
                ,Sk11-Sk21)*...
            exp(ui*pi*r/lambda*((tk*(1-0.5*
                sigmak_ant)-csi+1)^2/(1-tk
```

```
                        ^2-0.5*sigmak_ant)))*exp(-ui*zm(
                        j)*2*pi/lambda))+...
343                     (cf_sigma_ant(k).*complex(Ck22-Ck12,
                        Sk12-Sk22)*...
                        exp(ui*pi*r/lambda*((tk*(1-0.5*
                        sigmak_ant)-csi-1)^2/(1-tk
                        ^2-0.5*sigmak_ant))))*exp(ui*zm(
                        j)*2*pi/lambda)); % [V/m]
345                 else
                        Efs(k)= (0.5*cf*sqrt(r*lambda/(2*(1-
                        tk^2-0.5*sigmak_image)))*sth(k))
                        *...
347                     (cf_sigma_image(k).*(complex(Ck21-
                        Ck11,Sk11-Sk21)*...
                        exp(ui*pi*r/lambda*((tk*(1-0.5*
                        sigmak_image)-csi+1)^2/(1-tk
                        ^2-0.5*sigmak_image)))*exp(-ui*
                        zm(j-(N/2))*2*pi/lambda))+...
349                     (cf_sigma_image(k).*complex(Ck22-
                        Ck12,Sk12-Sk22)*...
                        exp(ui*pi*r/lambda*((tk*(1-0.5*
                        sigmak_image)-csi-1)^2/(1-tk
                        ^2-0.5*sigmak_image))))*exp(ui*
                        zm(j-(N/2))*2*pi/lambda)); % [V/
                        m]
351                 end
            else
353                 Efs(k)=Efr(k);

355             end
        end
357     Efs_tot = Efs_tot + Efs;
        end
359   Efs=Efs_tot;
      end
361 end
    %
363 figure(4)
    hold on
365 grid on
    box on
367 x=linspace(0,360,length(Efs));
    plot(x, 20*log10(abs(Efs)),'b-','Linewidth',2.5)
```

```
plot(x, 20*log10(abs(Efr)),'r-','Linewidth',2.5)
xlim([0 180])
ylim([max(20*log10(abs(Efr)))-30 max(20*log10(abs(Efr)))
    +5])
legend('E_{3t}','E_{2t}','Location','Best')

%integrals--------------------------%
x=linspace(0,pi,length(Efs)/2);     %
y=abs(Efs(1:180)).^2/Z0.*sin(x)*r^2;%
int_Efs = 2*pi*trapz(x,y);          %
y=abs(Efr(1:180)).^2/Z0.*sin(x)*r^2;%
int_Efr = 2*pi*trapz(x,y);          %
fprintf('int_Efs = %.6f\n',int_Efs);%
fprintf('int_Efr = %.6f\n',int_Efr);%
%----------------------------------%

if (strcmp(fstep_type,'e2t') == 1)
    f_step=fstep_uni(Efr,thetaM);
else
    f_step=fstep_e3t(Efs,thetaM);
end

DE2 = (abs(Efs).^2-abs(Efr).^2)/Z0;

% Gives output results in the required order
if strcmpi(first,'DE2');
    if strcmpi(second,'Efs');
        A=DE2;
        B=Efs;
        C=Efr;
        return
    else
        A=DE2;
        B=Efr;
        C=Efs;
        return
    end
end

if strcmpi(first,'Efs');
    if strcmpi(second,'DE2');
        A=Efs;
        B=DE2;
```

```
411             C=Efr;
                return
413         else
                A=Efs;
415             B=Efr;
                C=DE2;
417             return
            end
419 end

421 if strcmpi(first,'Efr');
        if strcmpi(second,'DE2');
423             A=Efr;
                B=DE2;
425             C=Efs;
                return
427         else
                A=Efr;
429             B=Efs;
                C=DE2;
431             return
            end
433 end

435

437 error('Incorrect output order definition:\n%s, %s, %s\n'
        ,first, second, third);

439

441 return;
```

A.3 EfieldLinearCurrentShift.m

```
% Computes the complex amplitude of electric field radiated by a linear
% constant current according to Fresnel and Fraunhofer approximation.
%
% Current distribution:
%       Je=I0*exp(-i (2*pi/lambda)*cos(thetaM)*z )
           -D/2<= z <= D/2
%
%
%                              /|\ axis z
%                               |
%                               +  D/2
%                               |
%                               |
%                               +--->
%                               |
%                               |
%                               + -D/2
%                               |
%
% I0 ................. current [A]
% theta, r ........... angle [deg] with respect to z positive axis, and
%                      distance [m] defining position at wich the
%                      radiated field will be computed
% D .................. current length [m]
% lambda  ............ vawelength [m];
% thetaM  ............ position (theta [deg]) of pattern peak;
%
% Efs ................ Fresnel approximation of electric field [V/m] at
%                      distance r and angular position theta
% Efr ................ Fraunhofer  approximation of electric field [V/m] at
%                      distance r and angular position theta
% DE2=(|Efs|^2-|Efr|^2)/Z0
```

```
%
% output: A,B,C correspond to DE2, Efs, Efr as specified by the input
% variables first, second, third
%
% The electric fieed in Fresnel region (Efs) is based on Fresnel (C, S)
% integrals. The computation of such a integrals is made exactly (by means
% of function Fresenl.m) or by an approximated formula (function
% FresnelApprox.m).
% The input variable FresnelComp select the Fresnel integral computation
% according to:
% FresnelComp='exact'  --> exact Fresnel integral computation
% FresnelComp='approx' --> approximated Fresnel integral computation
%
%
%


function [A,B,C,f_step] = EfieldLinearCurrent_shift(FresnelComp,I0,theta,...
r,D,ff_filename,lambda,h0,m2s,current_type,shield,fstep_type,thetaM,first,second,third)

c0 = 299.792458;      % free space light speed  [mm/ns]
Z0 = 4*pi*c0*10^-1;  % free space impedance [ohm] (approx. Z0=120*pi)

ui   = complex(0,1);  % imaginary unit
d2r = pi/180;         % deg to radian
method_sel = 2;       % 1 = element length fixed = lambda/2;
                      % 2 = elements length variable and related to N
D_tsc = antenna_D();
N=6;
% index = 2; %1:N=4; 2:N=6; 3:N=8; 4:N=10; 5:N=12; 6:N=14; 7:N=16.
```

```
% N=pair(index,1);

X=0.9;
if method_sel == 1
    %method 1
    elem_length=lambda/2;
    d_lambda=((D_tsc/lambda)-X)/(N-1);
    d=d_lambda*lambda;
    pos_z = linspace(0,d*(N-1),N)-d*(N-1)/2;
else
    %method 2
    pos_z = linspace(-D_tsc/2,D_tsc/2,N+1);
    elem_length=abs(pos_z(1)-pos_z(2));
    d=abs((pos_z(1)+pos_z(2))/2-(pos_z(2)+pos_z(3))/2);
    d_lambda=d/lambda;
end

%tsc coefficients normalization parameter
%normp = normalization_parameter(I0,theta,50,D,D_tsc,
    lambda,m2s,thetaM,N,d,method_sel,elem_length);
normp_efr = normalization_pirr(I0,theta,50,D_tsc,lambda,
    m2s,thetaM,N,d,method_sel,elem_length,FresnelComp,
    ff_filename,current_type,shield);

coeff = coefficient_tsc(N,m2s);
coeff=normp_efr*(coeff/max(coeff));
cf = -ui*Z0/(2*r*lambda)*I0*coeff*exp(-ui*2*pi*r/lambda)
    ; % Common factor

t  =  cos(d2r*theta);
csi=  cos(d2r*thetaM);

sth=  sin(d2r*theta);

ntheta=length(theta);

if shield == 0 %non viene simulata la presenza dello
    schermo
    x0=0;       %distance between the antenna and the
        shield
    beta=x0/r;
%     sigma_ant=beta^2-(2*beta*sqrt(1-t.^2));
%     sigma_image=beta^2+(2*beta*sqrt(1-t.^2));
```

```
else
    N=2*N;
    x0=0.054;   %distance between the antenna and the
        shield
    beta=x0/r;
    sigma_ant=beta^2-(2*beta*sqrt(1-t.^2));
    sigma_image=beta^2+(2*beta*sqrt(1-t.^2));
    cf_sigma_ant=exp(-ui*2*pi*r*((1/2)*sigma_ant-(1/8)*
        sigma_ant.^2)/lambda);
    %(-1)-->phase shift of 180 deg for the image (exp(j*
        pi))
    cf_sigma_image=(-1)*exp(-ui*2*pi*r*((1/2)*
        sigma_image-(1/8)*sigma_image.^2)/lambda);

end

if method_sel == 1
    %element length = lambda/2
    for j=1:length(pos_z)
        z1(j)=pos_z(j)-(elem_length/2);
        z2(j)=pos_z(j)+(elem_length/2);
        zm(j)=0.5*(z1(j)+z2(j));
        zd(j)=0.5*(z2(j)-z1(j));
    end
else
    %element length variable
    for j=1:length(pos_z)-1
        z1(j)=pos_z(j);
        z2(j)=pos_z(j+1);
        zm(j)=0.5*(z1(j)+z2(j));
        zd(j)=0.5*(z2(j)-z1(j));
    end
end
%z1=zm-zd
%z2=zm+zd

x=1;
y=1;
for i=1:2*length(z1)
    if mod(i,2) == 0
        z(i)=z2(x);
        x=x+1;
    else
```

```
138         z(i)=z1(y);
            y=y+1;
140     end
    end
142
    % preallocation to increase speed
144 %Efr=complex(10^-100,10^-100)*ones(1,ntheta);
    Efr=complex(10^-100,10^-100)*zeros(1,ntheta);
146 Efr_ant=Efr;
    Eft_image=Efr;
148 Efr_tot=Efr;
    Efs=Efr;
150 Efs_tot=Efr;
    DE2=Efr;
152
    %FRAUNHOFER
154 if(strcmp(current_type,'const') == 1)
        if(shield == 0) %corrente costante, senza schermo
156         for j=1:N
                rp=exp((ui*zm(j)*(t-csi)*2*pi)./lambda);
158             Efr=2*zd(j)*cf(j).*rp.*sth.*sinc(2*zd(j)*(t-
                    csi)/lambda);
                Efr_tot = Efr_tot+Efr;
160         end
        Efr=Efr_tot;
162     else  %corrente costante + schermo
            for j=1:N
164             if j <= N/2
                    rp_ant=cf_sigma_ant.*exp((ui*zm(j)*(t
                        .*(1-0.5*sigma_ant)-csi)*2*pi)./
                        lambda);
166                 Efr=zd(j)*cf(j).*sth.*(rp_ant.*sinc(2*zd
                        (j)*(t.*(1-0.5*sigma_ant)-csi)/
                        lambda));
                    Efr_tot = Efr_tot+Efr;
168             else
                    rp_image=cf_sigma_image.*exp((ui*zm(j-(N
                        /2))*(t.*(1-0.5*sigma_image)-csi)*2*
                        pi)./lambda);
170                 Efr=zd(j-(N/2))*cf(j-(N/2)).*sth.*(
                        rp_image.*sinc(2*zd(j-(N/2))*(t
                        .*(1-0.5*sigma_image)-csi)/lambda));
                    Efr_tot = Efr_tot+Efr;
```

```
172             end
            end
174     Efr=Efr_tot;
        end
176 else
        if(shield == 0) %corrente sinusoidale, senza schermo
178         for j=1:N
                rp1=exp((ui*zm(j)*(t-csi+1)*2*pi)./lambda).*
                    exp(-ui*zm(j)*2*pi/lambda);
180             rp2=exp((ui*zm(j)*(t-csi-1)*2*pi)./lambda).*
                    exp(ui*zm(j)*2*pi/lambda);
                Efr=zd(j)*cf(j).*sth.*((rp1.*sinc(2*zd(j)*(t
                    -csi+1)/lambda))+(rp2.*sinc(2*zd(j)*(t-
                    csi-1)/lambda)));
182             Efr_tot = Efr_tot+Efr;
            end
184         Efr=Efr_tot;
        else   %corrente sinusoidale + schermo
186         for j=1:N
                if j <= N/2
188                 rp1_ant=cf_sigma_ant.*exp((ui*zm(j)*(t
                        .*(1-0.5*sigma_ant)-csi+1)*2*pi)./
                        lambda).*exp(-ui*zm(j)*2*pi/lambda);
                    rp2_ant=cf_sigma_ant.*exp((ui*zm(j)*(t
                        .*(1-0.5*sigma_ant)-csi-1)*2*pi)./
                        lambda).*exp(ui*zm(j)*2*pi/lambda);
190                 Efr=zd(j)*cf(j).*sth.*((rp1_ant.*sinc(2*
                        zd(j)*(t.*(1-0.5*sigma_ant)-csi+1)/
                        lambda))+(rp2_ant.*sinc(2*zd(j)*(t
                        .*(1-0.5*sigma_ant)-csi-1)/lambda)))
                        ;
                    Efr_tot = Efr_tot+Efr;
192             else
                    rp1_image=cf_sigma_image.*exp((ui*zm(j-(
                        N/2))*(t.*(1-0.5*sigma_image)-csi+1)
                        *2*pi)./lambda).*exp(-ui*zm(j-(N/2))
                        *2*pi/lambda);
194                 rp2_image=cf_sigma_image.*exp((ui*zm(j-(
                        N/2))*(t.*(1-0.5*sigma_image)-csi-1)
                        *2*pi)./lambda).*exp(ui*zm(j-(N/2))
                        *2*pi/lambda);
                    Efr=zd(j-(N/2))*cf(j-(N/2)).*sth.*((
                        rp1_image.*sinc(2*zd(j-(N/2))*(t
```

```
                    .*(1-0.5*sigma_image)-csi+1)/lambda)
                    )+(rp2_image.*sinc(2*zd(j-(N/2))*(t
                    .*(1-0.5*sigma_image)-csi-1)/lambda)
                    ));
196                 Efr_tot = Efr_tot+Efr;
                end
198         end
        Efr=Efr_tot;
200     end
    end
202 %
    %FRESNEL
204 if (strcmp(current_type,'const') == 1) % CONSTANT
        CURRENT
        if(shield == 0) %constant current without shield
206         for j=1:N
                for k=1:ntheta
208             tk=t(k);
                if abs(tk)~=1
210                 % Fresnel integrals
                    k1c=sqrt(2*r/(lambda*(1-tk^2)));
212                 k1=k1c*((1-tk^2)*(z1(j)/(r))-(tk-csi));
                    k2=k1c*((1-tk^2)*(z2(j)/(r))-(tk-csi));
214                 if strcmpi(FresnelComp,'approx')
                        [Ck1,Sk1] = FresnelApprox(k1);
216                     [Ck2,Sk2] = FresnelApprox(k2);
                    else
218                     [Ck1,Sk1] = Fresnel(k1);
                        [Ck2,Sk2] = Fresnel(k2);
220                 end
                    %Fresnel field
222                 Efs(k)= (cf(j)*sqrt(r*lambda/(2*(1-tk^2)
                        ))*sth(k)*...
                        complex(Ck2-Ck1,Sk1-Sk2)*...
224                     exp(ui*pi*r/lambda*(-2+(tk-csi)
                            ^2/(1-tk^2)))); % [V/m]
                else
226                 Efs(k)=Efr(k);
                end
228          end
                Efs_tot = Efs_tot + Efs;
230         end
        Efs=Efs_tot;
```

```
232     else %constant current with shield
          for j=1:N
234             for k=1:ntheta
                tk=t(k);
236             sigmak_ant=sigma_ant(k);
                sigmak_image=sigma_image(k);
238             if abs(tk)~=1
                    if j <= N/2
240                     % Fresnel integrals
                        k1c=sqrt(2*r/(lambda*(1-tk^2-0.5*
                            sigmak_ant)));
242                     k1=k1c*((1-tk^2-0.5*sigmak_ant)*(z1(
                            j)/(r))-(tk*(1-0.5*sigmak_ant)-
                            csi));
                        k2=k1c*((1-tk^2-0.5*sigmak_ant)*(z2(
                            j)/(r))-(tk*(1-0.5*sigmak_ant)-
                            csi));
244                 else
                        k1c=sqrt(2*r/(lambda*(1-tk^2-0.5*
                            sigmak_image)));
246                     k1=k1c*((1-tk^2-0.5*sigmak_image)*(
                            z1(j-(N/2))/(r))-(tk*(1-0.5*
                            sigmak_image)-csi));
                        k2=k1c*((1-tk^2-0.5*sigmak_image)*(
                            z2(j-(N/2))/(r))-(tk*(1-0.5*
                            sigmak_image)-csi));
248                 end
                    if strcmpi(FresnelComp,'approx')
250                     [Ck1,Sk1] = FresnelApprox(k1);
                        [Ck2,Sk2] = FresnelApprox(k2);
252                 else
                        [Ck1,Sk1] = FresnelApprox(k1);
254                     [Ck2,Sk2] = FresnelApprox(k2);
                    end
256                 %Fresnel field
                    if j <= N/2
258                     Efs(k)= (0.5*cf(j)*sqrt(r*lambda
                            /(2*(1-tk^2-0.5*sigmak_ant)))*
                            sth(k))*...
                        (cf_sigma_ant(k).*(complex(Ck2-Ck1,
                            Sk1-Sk2)*...
260                     exp(ui*pi*r/lambda*((tk*(1-0.5*
                            sigmak_ant)-csi)^2/(1-tk^2-0.5*
```

```
                        sigmak_ant))))); % [V/m]
                else
262                     Efs(k)= (0.5*cf(j-(N/2))*sqrt(r*
                            lambda/(2*(1-tk^2-0.5*
                            sigmak_image)))*sth(k))*...
                        (cf_sigma_image(k).*(complex(Ck2-Ck1
                            ,Sk1-Sk2)*...
264                     exp(ui*pi*r/lambda*((tk*(1-0.5*
                            sigmak_image)-csi)^2/(1-tk
                            ^2-0.5*sigmak_image))))); % [V/m
                            ]
                end
266         else
                Efs(k)=Efr(k);
268
            end
270     end
        Efs_tot = Efs_tot + Efs;
272     end
    Efs=Efs_tot;
274 end
  %SINUSOIDAL CURRENT
276 else %sinusoidal current without shield
    if(shield == 0)
278     for j=1:N
            for k=1:ntheta
280             tk=t(k);
                if abs(tk)~=1
282                 % Fresnel integrals
                    k1c=sqrt(2*r/(lambda*(1-tk^2)));
284
                    k11=k1c*((1-tk^2)*(z1(j)/(r))-(tk-
                        csi+1));
286                 k21=k1c*((1-tk^2)*(z2(j)/(r))-(tk-
                        csi+1));
                    k12=k1c*((1-tk^2)*(z1(j)/(r))-(tk-
                        csi-1));
288                 k22=k1c*((1-tk^2)*(z2(j)/(r))-(tk-
                        csi-1));

290                 if strcmpi(FresnelComp,'approx')
                        [Ck11,Sk11] = FresnelApprox(k11)
                            ;
```

```
292                         [Ck21,Sk21] = FresnelApprox(k21)
                                ;
                            [Ck12,Sk12] = FresnelApprox(k12)
                                ;
294                         [Ck22,Sk22] = FresnelApprox(k22)
                                ;
                        else
296                         [Ck11,Sk11] = FresnelApprox(k11)
                                ;
                            [Ck21,Sk21] = FresnelApprox(k21)
                                ;
298                         [Ck12,Sk12] = FresnelApprox(k12)
                                ;
                            [Ck22,Sk22] = FresnelApprox(k22)
                                ;
300                     end
                        %Fresnel field
302                     Efs(k)= (0.5*cf(j)*sqrt(r*lambda
                            /(2*(1-tk^2)))*sth(k))*...
                            ((complex(Ck21-Ck11,Sk11-Sk21)
                                *...
304                         exp(ui*pi*r/lambda*((tk-csi+1)
                                ^2/(1-tk^2)))*exp(-ui*zm(j)
                                *2*pi/lambda))+...
                            (complex(Ck22-Ck12,Sk12-Sk22)
                                *...
306                         exp(ui*pi*r/lambda*((tk-csi-1)
                                ^2/(1-tk^2))))*exp(ui*zm(j)
                                *2*pi/lambda)); % [V/m]
                    else
308                     Efs(k)=Efr(k);
                    end
310             end
            Efs_tot = Efs_tot + Efs;
312         end
        Efs=Efs_tot;
314     else %sinusoidal current with shield
          for j=1:N
316             for k=1:ntheta
                tk=t(k);
318             sigmak_ant=sigma_ant(k);
                sigmak_image=sigma_image(k);
320             if abs(tk)~=1
```

```
                if j <= N/2
322                 % Fresnel integrals
                    k1c=sqrt(2*r/(lambda*(1-tk^2-0.5*
                        sigmak_ant)));
324                 k11=k1c*((1-tk^2-0.5*sigmak_ant)*(z1
                        (j)/(r))-(tk*(1-0.5*sigmak_ant)-
                        csi+1));
                    k21=k1c*((1-tk^2-0.5*sigmak_ant)*(z2
                        (j)/(r))-(tk*(1-0.5*sigmak_ant)-
                        csi+1));
326                 k12=k1c*((1-tk^2-0.5*sigmak_ant)*(z1
                        (j)/(r))-(tk*(1-0.5*sigmak_ant)-
                        csi-1));
                    k22=k1c*((1-tk^2-0.5*sigmak_ant)*(z2
                        (j)/(r))-(tk*(1-0.5*sigmak_ant)-
                        csi-1));
328             else
                    k1c=sqrt(2*r/(lambda*(1-tk^2-0.5*
                        sigmak_image)));
330                 k11=k1c*((1-tk^2-0.5*sigmak_image)*(
                        z1(j-(N/2))/(r))-(tk*(1-0.5*
                        sigmak_image)-csi+1));
                    k21=k1c*((1-tk^2-0.5*sigmak_image)*(
                        z2(j-(N/2))/(r))-(tk*(1-0.5*
                        sigmak_image)-csi+1));
332                 k12=k1c*((1-tk^2-0.5*sigmak_image)*(
                        z1(j-(N/2))/(r))-(tk*(1-0.5*
                        sigmak_image)-csi-1));
                    k22=k1c*((1-tk^2-0.5*sigmak_image)*(
                        z2(j-(N/2))/(r))-(tk*(1-0.5*
                        sigmak_image)-csi-1));
334             end
                if strcmpi(FresnelComp,'approx')
336                 [Ck11,Sk11] = FresnelApprox(k11);
                    [Ck21,Sk21] = FresnelApprox(k21);
338                 [Ck12,Sk12] = FresnelApprox(k12);
                    [Ck22,Sk22] = FresnelApprox(k22);
340             else
                    [Ck11,Sk11] = FresnelApprox(k11);
342                 [Ck21,Sk21] = FresnelApprox(k21);
                    [Ck12,Sk12] = FresnelApprox(k12);
344                 [Ck22,Sk22] = FresnelApprox(k22);
                end
```

```
346                     %Fresnel field
                        if j <= N/2
348                         Efs(k)= (0.5*cf(j)*sqrt(r*lambda
                                /(2*(1-tk^2-0.5*sigmak_ant)))*
                                sth(k))*...
                            (cf_sigma_ant(k).*(complex(Ck21-Ck11
                                ,Sk11-Sk21)*...
350                             exp(ui*pi*r/lambda*((tk*(1-0.5*
                                sigmak_ant)-csi+1)^2/(1-tk
                                ^2-0.5*sigmak_ant)))*exp(-ui*zm(
                                j)*2*pi/lambda))+...
                            (cf_sigma_ant(k).*complex(Ck22-Ck12,
                                Sk12-Sk22)*...
352                             exp(ui*pi*r/lambda*((tk*(1-0.5*
                                sigmak_ant)-csi-1)^2/(1-tk
                                ^2-0.5*sigmak_ant))))*exp(ui*zm(
                                j)*2*pi/lambda)); % [V/m]
                        else
354                         Efs(k)= (0.5*cf(j-(N/2))*sqrt(r*
                                lambda/(2*(1-tk^2-0.5*
                                sigmak_image)))*sth(k))*...
                            (cf_sigma_image(k).*(complex(Ck21-
                                Ck11,Sk11-Sk21)*...
356                             exp(ui*pi*r/lambda*((tk*(1-0.5*
                                sigmak_image)-csi+1)^2/(1-tk
                                ^2-0.5*sigmak_image)))*exp(-ui*
                                zm(j-(N/2))*2*pi/lambda))+...
                            (cf_sigma_image(k).*complex(Ck22-
                                Ck12,Sk12-Sk22)*...
358                             exp(ui*pi*r/lambda*((tk*(1-0.5*
                                sigmak_image)-csi-1)^2/(1-tk
                                ^2-0.5*sigmak_image))))*exp(ui*
                                zm(j-(N/2))*2*pi/lambda)); % [V/
                                m]
                        end
360             else
                    Efs(k)=Efr(k);
362
                end
364         end
            Efs_tot = Efs_tot + Efs;
366         end
        Efs=Efs_tot;
```

```
368   end
  end
370 %
  figure(4)
372 hold on
  grid on
374 box on
  x=linspace(0,360,length(Efs));
376 plot(x, 20*log10(abs(Efs)/max(abs(Efs))),'m-','Linewidth
      ',2.5)
  plot(x, 20*log10(abs(Efr)/max(abs(Efs))),'k-','Linewidth
      ',2.5)
378 xlim([0 180])
  ylim([-50 5])
380 legend('E_{3t}','E_{2t}')
  %legend('Fraunhofer DT','Fraunhofer DT','Fresnel DT','
      Fresnel DT','E_{3t}','E_{2t}','Location','Best')
382 title1=sprintf('R_0 = %d dB',m2s);
  title(title1)
384
  max_E3t=max(abs(Efs));
386 figure(10)
  hold on
388 grid on
  box on
390 plot(x, 20*log10(abs(Efs)/max_E3t),'m-','Linewidth',2.5)
  xlim([0 90])
392 ylim([-30 5])
  title(title1)
394
  if (strcmp(fstep_type,'e2t') == 1)
396     f_step=fstep_prova(Efr,m2s);
  else
398     f_step=fstep_e3t(Efs,m2s);
  end
400 %integrals--------------------------%
  x=linspace(0,pi,length(Efs)/2);      %
402 y=abs(Efs(1:180)).^2/Z0.*sin(x)*r^2;%
  int_Efs = 2*pi*trapz(x,y);           %
404 y=abs(Efr(1:180)).^2/Z0.*sin(x)*r^2;%
  int_Efr = 2*pi*trapz(x,y);           %
406 fprintf('int_Efs = %.6f\n',int_Efs);%
  fprintf('int_Efr = %.6f\n',int_Efr);%
```

```
%-------------------------------------%
DE2 = (abs(Efs).^2-abs(Efr).^2)/Z0;

% Gives output results in the required order
if strcmpi(first,'DE2');
    if strcmpi(second,'Efs');
        A=DE2;
        B=Efs;
        C=Efr;
        return
    else
        A=DE2;
        B=Efr;
        C=Efs;
        return
    end
end

if strcmpi(first,'Efs');
    if strcmpi(second,'DE2');
        A=Efs;
        B=DE2;
        C=Efr;
        return
    else
        A=Efs;
        B=Efr;
        C=DE2;
        return
    end
end

if strcmpi(first,'Efr');
    if strcmpi(second,'DE2');
        A=Efr;
        B=DE2;
        C=Efs;
        return
    else
        A=Efr;
        B=Efs;
        C=DE2;
        return
```

```
        end
452 end

454

456 error('Incorrect output order definition:\n%s, %s, %s\n'
        ,first, second, third);

458

460 return;
```

A.4 FieldCorrections.m

```
function [A,B,C] = FieldCorrections(r,theta,FresnelComp,...
                                    I0,D,ff_filename,
                                        lambda,h0,m2s_dB,
                                        Power,
                                        current_type,
                                        shield,
                                        delta2_flag,
                                        fstep_type,
                                        GMAXlin,tilt,
                                        sin_exp,first,
                                        second,third)


% According to "A Fraunhofer-based Approach for the Assessment of the Field
% Radiated in the Fresnel Region of an Antenna", APWL, this routine computes
% the  squared electric field additive term given by equation 8 and 10.
% The computation is carried out at distance Position.r and angle
% Position.Theta.
%
% The parameters of the linear current distribution (equation 3) are:
% I0 ..................  current [A]
% D ...................  current length [m]
% lambda  .............  wavelength [m];
% thetaM  .............  position (theta [deg]) of pattern peak;


r2d=180/pi;
d2r=pi/180;

c0 = 299.792458*10^6;% free space light speed  [mm/ns]
Z0 = 4*pi*299.792458*10^-1;  % free space impedance [ohm] (approx. Z0=120*pi)

```

```
costhetaM = cos(d2r*(90+tilt));

if h0 == 0
    theta1    = r2d*acos(costhetaM +lambda/D);
    theta2    = r2d*acos(costhetaM -lambda/D);
else
    [pair, D_lambda]=min_quadrati_tsc_sel(m2s_dB);
    theta1    = r2d*acos(costhetaM +lambda/D);
    theta2    = r2d*acos(costhetaM -lambda/D);
end


    %thetaDE2  = [theta1 theta2 (90+tilt) theta];  %old
        version
    thetaDE2  = theta; %modificato 31 08
    %thetaDE2  = [(90+tilt) theta1 theta2 theta];


%thetaDE2  = [(90+tilt) theta1 theta2 theta];

    if(h0 == 0)
         [DELTA,Efs,Efr,f_step] = EfieldLinearCurrent(
             FresnelComp,...
                      I0, .......... complex current
                          amplitude [A]
                      thetaDE2,......theta angle [deg]
                      r,... distance [m]
                      D,............ current length [m]
                      ff_filename,.........
                      lambda,....... wavelength [mm]
                      current_type,......current can be
                          constant or sinusoidal
                      shield,........the simulation can
                          include or not the presence of
                           the shield
                      fstep_type,.......f_step can be
                          based on e3t or e2t
                      (90+tilt),....... theta position [
                          deg] of pattern peak
                      'DE2',........ first output data
                          is |Efs|^2-|Efr|^2
                      'Efs',........ second output data
                          is Efs
```

```
58                      'Efr');%..... third output data
                            is Efr
    else
60          [DELTA,Efs,Efr,f_step] =
                EfieldLinearCurrent_shift(FresnelComp,...
                        I0, ......... complex current
                            amplitude [A]
62                      thetaDE2,.....theta angle [deg]
                        r,... distance [m]
64                      D,............ current length [m]
                        ff_filename,.....
66                      lambda,....... wavelength [mm]
                        h0,...........shift
68                      m2s_dB,..........major-to-side-
                            lobe ratio [dB]
                        current_type,.....current can be
                            constant or sinusoidal
70                      shield,...........the simulation
                            can include or not the
                            presence of the shield
                        fstep_type,.......f_step can be
                            based on e3t or e2t
72                      (90+tilt),....... theta position [
                            deg] of pattern peak
                        'DE2',........ first output data
                            is |Efs|^2-|Efr|^2
74                      'Efs',........ second output data
                            is Efs
                        'Efr');%..... third output data
                            is Efr
76  end

78
    %deltaP_0 = DELTA(4:end);  %old version %first
        additive incremental term    modificato 22 07
80  deltaP_0 = DELTA; %modificato 31 08

82
      [DELTA_MIN idx] = min(DELTA(1:181));
84    [DELTA_MAXLF idx_left] = max(DELTA(1:idx));
      [DELTA_MAXRG idx_right] = max(DELTA(idx:181));
86    idx_right = idx_right + idx -1;
```

```
88
     %deltaP_1 = (abs(DELTA(1))-(abs(DELTA(2))+abs(DELTA
         (3))))/3;%old
90   %version %second additive incremental term %
         modificato 31 08

92   deltaP_1 = (abs(DELTA_MIN)-(abs(DELTA_MAXRG)+abs(
         DELTA_MAXLF)))/3;

94   AF_s=AF_shield;
%    if h0 == 0
96 %      f_step=fstep_uni(ff_filename,tilt);
%    else
98 %      f_step=fstep_prova(ff_filename,m2s_dB);
%    end
100
     if strcmp(delta2_flag,'sin') == 1
102       deltaP_2   = deltaP_1 * sintilt(theta,tilt,
              sin_exp);%  Attenzione!! --> sintilt eleva
              già al quadrato!!
     end
104  if strcmp(delta2_flag,'sin_step') == 1
       deltaP_2   = deltaP_1 * (f_step).^2.*sintilt(
            theta,tilt,sin_exp);
106  end
     if strcmp(delta2_flag,'tot') == 1
108     deltaP_2   = deltaP_1 *(AF_s.*f_step).^2.*sintilt
            (theta,tilt,sin_exp);
     end
110
     figure(6)
112  hold on
     grid on
114  plot(theta, ((AF_s).^2),'r');
     plot(theta, ((f_step).^2),'b');
116  plot(theta, ((sin(theta*pi/180)).^2),'m');
     plot(theta, ((AF_s.*sin(theta*pi/180)).^2),'g')
118  plot(theta, ((AF_s.*f_step.*sin(theta*pi/180)).^2),'k
         ','Linewidth',2);
     xlim([0 180])
120  xlabel('\theta')
     legend('AF_{shield}','F_{step}','SIN','SIN*AF_{shield
         }','TOTAL')
```

```
122     title1=sprintf('R0 = %d dB', m2s_dB);
        title(title1)
124
      % Gives output results in the required order
126 if strcmpi(first,'DE0');
        if strcmpi(second,'DE1');
128         A=deltaP_0;
            B=deltaP_1;
130         C=deltaP_2;
            return
132     else
            A=deltaP_0;
134         B=deltaP_2;
            C=deltaP_1;
136         return
        end
138 end

140 if strcmpi(first,'DE1');
        if strcmpi(second,'DE0');
142         A=deltaP_1;
            B=deltaP_0;
144         C=deltaP_2;
            return
146     else
            A=deltaP_1;
148         B=deltaP_2;
            C=deltaP_0;
150         return
        end
152 end

154 if strcmpi(first,'DE2');
        if strcmpi(second,'DE0');
156         A=deltaP_2;
            B=deltaP_0;
158         C=deltaP_1;
            return
160     else
            A=deltaP_2;
162         B=deltaP_1;
            C=deltaP_0;
164         return
```

```
        end
166 end

168

170 error('Incorrect output order definition:\n%s, %s, %s\n'
        ,first, second, third);

172

174 return;
```

A.5 FresnelApprox.m

```
% Approximated Fresnel integrals with real (-inf, +inf)
    argument.
%
% Fresnel Integral C(x) and S(x) are defined by:
%          x
%         /
% C(x)= | cos( pi/2 * t^2) dt;
%       /
%       0
%
%          x
%         /
% S(x)= | sin( pi/2 * t^2) dt;
%       /
%       0
%
% The approximated formula which estimates C(x)+j S(x)
    is based on:
%
% 1) Taylor series at zero (i.e. summ of x^n)
                for |x|<=xtrs (treshold);
%    see M.Abramowitz, I Stegun, HandBook of
    Matehematical functions, Dover,
%    New York, ninth printing, 1970, pag. 301, eq.
    7.3.12 and 7.3.14
%
%
% 2) three terms series at infinity (i.e. summ of 1/x^n)
     for |x|>xtrs
%
%    To obtain this asymptotic formula the real axis
    integration path has
%    been deformed into complex plane path (first
    quadrant bisector and
%    hyperbola) leading to the following exact
    expression (valid for any x
%    not equal to zero):
%                          1+j
%    C(x)+jS(x)=sign(x)*----- +
%                           2
%                                    + inf
```

```
%                                    /
%                                   |          t
%            -x*exp(j*pi*x^2/2)*    | (------------+j) exp
    (-pi*x^2*t*sqrt(1+t^2) dt;
%                                   |   sqrt(1+t^2)
%                                  /
%                                  0
%
%    The above integral can be expressed in terms of a
    series expansion at
%    infinity. The series, truncated at the first three
    term gives:
%
%                           1+j
%    C(x)+jS(x)~ sign(x)*----- +
%                            2
%               - (1/pi*x)*exp(j*pi*x^2/2) * ( a*(1+15*a
    ^2*(-1+63*a^2))+
%                                              j*(1+3*a
    ^2*(-1+35*a^2)) )
%    where a=1/(pi*x^2)
%
%    The default set xtrs=2.6 and n0=17   guarantee an
    error
%    (max( abs(C(x)-C_approx(x)), abs(S(x)-S_approx(x))
    ) less than 10^-5.
%    Further values are given in the following table
%
%    ---------------------------
%    | xtrs | n0  |   max error |
%    ---------------------------
%    |  2.6 | 17  |    < 10^-5  |
%    |  3.2 | 26  |    < 10^-6  |
%
% Input:  x    (array)
% Output: C, S (arrays)
%
%

function [C,S] = FresnelApprox(x,xtrs,n0)

if nargin ==1
    % treshold between the two approximated expression
```

```
        xtrs= 2.6;
70
        % no. of element of the truncated series (n
            =0,1,2,..,no) of x^n series
72      n0  = 17;

74  else
        if nargin ~=3
76          error('Both treshold and no. of elements of
                truncated series must be given');
        end
78  end


80
    % imaginary unit
82  ui=complex(0,1);

84  % 1/pi
    pim1=1/pi;
86
    nx=length(x);
88
    % preallocation
90  C=zeros(1,nx);
    S=zeros(1,nx);
92
    for k=1:nx
94      xk   = x(k);
        xksign = sign(xk);
96
        % computes exp(ui*pi/2*xk^2)
98      m1    = round(xk/2);
        d1    = xk-2*m1;
100
        d1m1  = d1*m1;
102     m2    = round(d1m1);
        d2    = d1m1-m2;
104     expxk2 = exp(ui*pi/2*(d1^2+4*d2));

106     if abs(xk)<= xtrs

108         % EQ. 7.3.12 pag 301 Abramovitz, series
                truncated to
```

```
        % n0 terms (n0=0,1,2,...)
110     serie1=1;
        xk4pi2=xk^4*pi^2;
112     for j=n0:-1:1
            serie1=1- serie1* xk4pi2/((4*j-1)*(4*j+1));
114     end

116     % EQ. 7.3.14 pag 301 Abramovitz, series
            truncated to
        % n0 terms (n0=0,1,2,...)
118     serie2=1;
        for j=n0:-1:1
120         serie2=1- serie2* xk4pi2/((4*j+1)*(4*j+3));
        end
122     serie2=serie2*xk^2*pi/3;

124     CS=expxk2*xk*complex(serie1,-serie2);

126 else
        % asymptotic expansion
128     a=pim1/xk^2;
        b=pim1/xk;
130
        CS=xksign*complex(0.5,0.5)... first quadrant
            bisector path integration (analytical result
            )
132         -expxk2*b*complex(a*(1+15*a^2*(-1+63*a^2))
                ,...        first quadrant hyperbola path
                                (1+3*a^2*(-1+35*a
                                    ^2))); %
                                    integration (
                                    approximated
                                    computation)
134
    end
136
    C(k)=real(CS);
138 S(k)=imag(CS);

140 end

142 return;
```

Appendix B

NMEA frame analysis

B.1 main.m

```
clc
clear all
close all

% Import the GPS data frames that are in excel format
% The data are imported as .mat file
%[numeric_data,txt_data,all_data]=xlsread('
    name_of_the_file','sheet_of_the_excel_file');

%authors: Vanessa Bataller
%                   Alessandra Carta

gpsselected='nmea3';

switch gps_selected
    case 'nmea1'
        color='r-';
    case 'nmea2'
        color='b-';
    case 'nmea3'
        color='c-';
end

range='A1:U168727';

%[satelites,txt,all]=xlsread('gps_17mayo2012.xlsx','
    nmea1');
```

```
[satelites,txt,all]=xlsread('gps301102012.xlsx',
    gpsselected);
%[satelites,txt,all]=xlsread('prova_err.xlsx',
    gps_selected);

[m,n]=size(satelites);
%controllo aggiunto
%satelites=satelites(:,2:end);

%Inicialization of data matrix
matrizgpgga=zeros(1,6);
matrizgpgsvelev=zeros(1,32);
matrizgpgsvazim=zeros(1,32);
matrizgpgsvsnr=zeros(1,32);
matrizgpgsa=zeros(1,15);
conttrama1=1;

while  cont_trama1<m       %loop that reads every frame

    flag=0;
    trama=all(conttrama1,:);
    tramanum=satelites(conttrama1,:);
    tramatxt=txt(conttrama1,:);
    tipotrama=tramatxt(1);

    %NMEA frame GPGGA
    if strcmp(tipotrama,'GPGGA')==1

            flag=controlgpgga(tramatxt);
            if flag == 0
                [horanum,horastr,lat,long,posx,posy,alt
                    ]=...
                    tramagpgga(tramanum,tramatxt);
                conttrama1=conttrama1+1;
                %the data to be saved are the time,
                    latitude, longitude,
                %position x in UTM coordinates, position
                     y in UTM coordinates
                %and altitude
                vectorgga=[horanum lat long posx posy
                    alt];
                matrizgpgga=[matrizgpgga; vectorgga];
```

```
            else
64              cont_trama1=cont_trama1+1;
            end
66

68      %NMEA frame GPGSV
        else if strcmp(tipotrama,'GPGSV')==1
70          %The number of GSV frames (1 to 3) are read,
                 according to the
            %first field of the gsv frame
72
            flag=control_gpgsv(tramatxt);
74          if flag == 0

76              numtrama=tramanum(1);

78              %tramas=all(cont_trama1:cont_trama1+
                    numtrama-1,:);
                tramasnum=satelites(conttrama1:
                    conttrama1+numtrama-1,:);
80              [vectorelev,vectorazim, vectorsnr]=
                    tramagpgsv(tramasnum);
                conttrama1=conttrama1+numtrama;
82              %The data are saved in three matrix, one
                     for each parameter
                %(elevation, azimuth and SNR)
84              matrizgpgsvelev=[matrizgpgsvelev;
                    vectorelev];
                matrizgpgsvazim=[matrizgpgsvazim;
                    vectorazim];
86              matrizgpgsvsnr=[matrizgpgsvsnr;vectorsnr
                    ];
            else
88               conttrama1=conttrama1+1;
            end
90

92        %NMEA frame GPGSA
          else if strcmp(tipotrama,'GPGSA')==1
94
            flag=control_gpgsa(tramatxt);
96          if flag == 0
                [prn,pdop,hdop,vdop]=tramagpgsa(tramanum
```

```
                    );
98                  conttrama1=conttrama1+1;
                    vectorgsa=[prn,pdop,hdop,vdop];
100                 matrizgpgsa=[matrizgpgsa; vectorgsa];
                end
102                 conttrama1=conttrama1+1;
                end
104           %Other frames that are not analyzed
                else
106             conttrama1=conttrama1+1;
                end;
108         end;
      end;
110 end;

112 plottramagpgga(color,matrizgpgga(:,1),matrizgpgga(:,2)
      ,...
      matrizgpgga(:,3),matrizgpgga(:,4),matrizgpgga(:,5),
          matrizgpgga(:,6));
114 plottramagpgsv(color,matrizgpgga(:,1),matrizgpgsvelev
      ,...
      matrizgpgsvazim,matrizgpgsvsnr);
116 plottramagpgsa(color,matrizgpgsa(:,13),matrizgpgsa(:,14)
      ,matrizgpgsa(:,15));
```

B.2 GPGGA.m

```
function [horanum,horastr,lat,long,este,norte,alt]=
    trama_gpgga(tramanum,tramatxt)
%This function reads gpgga frames in vector format. It
    outputs the time,
%position and identifier of the satellites (PRN) used
    for the position
%calculation

    % zero padding added for time around midnight
    %tramanum_padded=sprintf('%06d',tramanum(1));
    %horanum=datenum(num2str(tramanum_padded),'HHMMSS');
    horanum=datenum(num2str(tramanum(1)),'HHMMSS');
    horastr=datestr(horanum,'HH:MM:SS');
    latdeg=floor(tramanum(2)/100); %latitude in degrees
    latmin=tramanum(2)-latdeg*100;
    lat=latdeg+latmin/60;
    if strcmp(tramatxt(4),'S')  %latitude N or S
        nor=0;
    else
        nor=1;
    end;
    longdeg=floor(tramanum(4)/100); %Longitude in
        degrees
    longmin=tramanum(4)-longdeg*100;
    long=longdeg+longmin/60;
    if strcmp(tramatxt(6),'W') %Longitude W or E
        long=-long;
    end;

    %the position in geodetic coordinates is converted
        into UTM coordinates
    [este,norte,huso]=geode2utm(long,lat,nor);
    alt=tramanum(9);    %altitude
```

B.3 GPGSA.m

```
function [prn,pdop,hdop,vdop]=trama_gpgsa(tramanum)
%this function reads the gpgsa frames and returns  the prn of the satellites seen, the pdop,
%hdop and vdop
for k=3:14
    if isnan(tramanum(k))==1
        prn(k-2)=0;
    else
        prn(k-2)=tramanum(k);
    end;
end;

pdop=tramanum(15);
hdop=tramanum(16);
vdop=tramanum(17);
```

B.4 GPGSV.m

```
function [vector_elev, vector_azim, vector_snr]=
    trama_gpgsv(tramasnum)
%This function reads the gsv frames (1 to 3 frames). It
    returns the PRN of the satellites,
%elevation, azimuth and SNR

[n,m]=size(tramasnum);
for k=1:n
    for j=1:m
        if isnan(tramasnum(n,m))==1
            tramasnum(n,m)=0;
        end;
    end;
end;

    %See the available satelliltes
    numsat=tramasnum(1,3); %number of satellites seen
    %The number of satellites by row of the gsv frames
        according to the row
    %is calculated

    [numsat1,numsat2,numsat3]=satporfila(numsat);


    %The data matrix are created
    vector_snr=zeros(1,32);
    vector_azim=zeros(1,32);
    vector_elev=zeros(1,32);

    %read the first row
    for t=0:numsat1-1
        prn=tramasnum(1,4+t*4); %fields 1, 5, 9 y 13
        elev=tramasnum(1,5+t*4);%fields 2, 6, 10y 14
        azim=tramasnum(1,6+t*4);%fields 3, ...
        snr=tramasnum(1, 7+t*4);%fields 4...

        %It looks for the PRN in the vector of index and
            fills in the
        %vector with the new PRN if it is a new value
        index_prn=devuelve(prn);

```

```
        vector_snr(index_prn)=snr;
39      vector_azim(index_prn)=azim;
        vector_elev(index_prn)=elev;
41  end;

43  if numsat2>0 %If there are two rows of gsv frames
        for t=0:numsat2-1
45          prn=tramasnum(2, 4+t*4);
            elev=tramasnum(2, 5+t*4);
47          azim=tramasnum(2, 6+t*4);
            snr=tramasnum(2, 7+t*4);
49          %Look for prn
            index_prn=devuelve(prn);
51          vector_snr(index_prn)=snr;
            vector_azim(index_prn)=azim;
53          vector_elev(index_prn)=elev;
        end;
55  end;
    if numsat3>0 %If there are three rows of gsv frames
57      for t=0:numsat3-1
            prn=tramasnum(3, 4+t*4);
59          elev=tramasnum(3, 5+t*4);
            azim=tramasnum(3, 6+t*4);
61          snr=tramasnum(3, 7+t*4);

63          %busca prn
            index_prn=devuelve(prn);
65          vector_snr(index_prn)=snr;
            vector_azim(index_prn)=azim;
67          vector_elev(index_prn)=elev;
        end;
69  end;
```

B.5 plotGPGGA.m

```
function plot_trama_gpgga(color,horanum,lat,long,posx,
    posy,alt)

offset=10; %to exclude the first data (incorrect)

stile=color;
figure(37)
hold on
grid on
box on
plot(horanum(offset:end),lat(offset:end),stile)
%increase the displayed precision on the plot
yt=get(gca,'YTick');
ylab=num2str(yt(:),15);
set(gca,'YTicklabel',ylab);
%
xlabel('Time UTC')
datetick('x','HH:MM')
ylabel('Latitude [deg]')

figure(38)
hold on
grid on
box on
plot(horanum(offset:end),long(offset:end),stile)
%increase the displayed precision on the plot
yt=get(gca,'YTick');
ylab=num2str(yt(:),15);
set(gca,'YTicklabel',ylab);
%
datetick('x','HH:MM')
xlabel('Time UTC')
ylabel('Longitude [deg]')

figure(39)
hold on
grid on
box on
plot(horanum(offset:end),alt(offset:end),stile)
%increase the displayed precision on the plot
yt=get(gca,'YTick');
```

```
ylab=num2str(yt(:),15);
set(gca,'YTicklabel',ylab);
%
datetick('x','HH:MM')
xlabel('Time UTC')
ylabel('Altitude [m]')

figure(40)
hold on
grid on
box on
plot(horanum(offset:end),posx(offset:end),stile)
%increase the displayed precision on the plot
yt=get(gca,'YTick');
ylab=num2str(yt(:),15);
set(gca,'YTicklabel',ylab);
%
datetick('x','HH:MM')
xlabel('Time UTC')
ylabel('Position East [m]')

figure(41)
hold on
grid on
box on
plot(horanum(offset:end),posy(offset:end),stile)
%increase the displayed precision on the plot
yt=get(gca,'YTick');
ylab=num2str(yt(:),15);
set(gca,'YTicklabel',ylab);
%
datetick('x','HH:MM')
xlabel('Time UTC')
ylabel('Position North [m]')


% figure(4)
% hold on
% grid on
% box on
% plot(str2num(datestr(horanum)),'-')
```

B.6 plotGPGSA.m

```
function plot_trama_gpgsa(color,pdop,hdop,vdop)

stile=strcat(color,'*');
figure(42)
hold on
grid on
box on
plot(hdop,stile)
xlabel('Time [s]')
ylabel('Horizontal Dilution of Precision [m]')

figure(43)
hold on
grid on
box on
plot(pdop,stile)
xlabel('Time [s]')
ylabel('Position Dilution of Precision [m]')

figure(44)
hold on
grid on
box on
plot(vdop,stile)
xlabel('Time [s]')
ylabel('Vertical Dilution of Precision [m]')
%
```

B.7 plotGPGSV.m

```
function plot_trama_gpgsv(color,horanum,elevation,
    azimuth, snr)

offset=10;
stile=color;
vector_prn=[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
     19 20 21 22 23 25 26 28 29 30 31 32 120 122];
%vector_int=[2 4 9 12 14 25 29 31];
vector_int=[1 3 6 7 8 11 16 18 19 21];

%control over the dimension of the time and snr vectors
dim_horanum=length(horanum);
dim_snr=size(snr,1);

if dim_horanum > dim_snr
    diff=dim_horanum-dim_snr;
    horanum_new=horanum(diff+1:end);
    snr_new=snr;
    azimuth_new=azimuth;
    elevation_new=elevation;
else
    diff=dim_snr-dim_horanum;
    snr_new=snr(diff+1:end,:);
    azimuth_new=azimuth(diff+1:end,:);
    elevation_new=elevation(diff+1:end,:);
    horanum_new=horanum;
end

%plot the snr for all the satellites
for i=1:length(vector_int)
    figure(i)
    hold on
    grid on
    box on
    index=devuelve(vector_int(i));
    plot(horanum_new(offset:end),snr_new(offset:end,
        index),stile,'Linewidth',1)
    %plot(horanum(offset:end),ones(1,length(horanum)-
        offset+1),'w')
    titolo=sprintf('PRN %d',vector_prn(index));
    filename=sprintf('D:\\ALE_iXem\\DOTTORATO\\
```

```
        GPS_spagna\\MATLAB\\gps_matlab\\
        figure_satellites\\SNR_time_%d.fig',vector_prn(
        index));
    title(titolo)
    ylim([0 60])
    xlabel('Time UTC')
    datetick('x','HH:MM')
    ylabel('SNR [dB]')
    saveas(gcf,filename)
end
%
for i=1:length(vector_int)
    figure(i+length(vector_int))
    hold on
    grid on
    box on
    index=devuelve(vector_int(i));
    plot(elevation_new(offset:end,index),snr_new(offset:
        end,index),stile,'Linewidth',1)
    titolo=sprintf('PRN %d',vector_prn(index));
    filename=sprintf('D:\\ALE_iXem\\DOTTORATO\\
        GPS_spagna\\MATLAB\\gps_matlab\\
        figure_satellites\\SNR_elevation_%d.fig',
        vector_prn(index));
    title(titolo)
    ylim([0 60])
    xlabel('Elevation [deg]')
    ylabel('SNR [dB]')
    saveas(gcf,filename)
end

for i=1:length(vector_int)
    figure(i+length(vector_int)*2)
    hold on
    grid on
    box on
    index=devuelve(vector_int(i));
    plot(azimuth_new(offset:end,index),snr_new(offset:
        end,index),stile,'Linewidth',1)
    titolo=sprintf('PRN %d',vector_prn(index));
    filename=sprintf('D:\\ALE_iXem\\DOTTORATO\\
        GPS_spagna\\MATLAB\\gps_matlab\\
        figure_satellites\\SNR_azimuth_%d.fig',
```

```
            vector_prn(index));
71      title(titolo)
        ylim([0 60])
73      xlabel('Azimuth [deg]')
        ylabel('SNR [dB]')
75      saveas(gcf,filename)
    end
```

Bibliography

[1] IEGMP, `http://www.iegmp.org.uk/report/text.htm`.

[2] P. Bernardi, M. Caragnaro, S. Pisa, and E. Piuzzi, *"Human exposure to radio base-station antennas in urban environment"*, IEEE Trans. Microwave Theory Tech., Nov. 2000.

[3] G. Lazzi and O. P. Gandhi, *"A mixed FDTD-integral equation approach for on-site safety assessment in complex electromagnetic environments"*, IEEE Trans. Antennas Propagat., Dec. 2000.

[4] A. Faraone, R.Y.S. Tay, K.H. Joyner, and Q. Balzano, *"Estimation of the average power density in the vicinity of cellular base-station collinear array antenna"*, IEEE Transaction VT, May 2000.

[5] R. Cicchetti and A. Faraone, *"Estimation of the peak power density in the vicinity of cellular and radio base station antennas"*, IEEE Trans EMC, May 2004.

[6] T. Kobayashi and T. Nojima, *"Estimation of electric field intensity in the Fresnel region of collinear array antennas"* IEICE, Trans. Communication., Jun 1994.

[7] S. Blanch, J. Romeu, and A. Cardama, *"Near-field in the vicinity of wireless base-station antennas: An exposure compliance approach"*, IEEE Trans. Antennas Propagat., May 2002.

[8] C.A. Balanis, "Antenna Theory. Analysis and Design", Arizona State University, John & Sons, 1989.

[9] D. Forigo, P. Gianola, R. Scotti, R. Vallauri, *"Measurements and Numerical Evaluation of the Electric Field in the Near Zone of Radio Base Station Antennas"*, IEEE Antennas Propagat. Int. Symposium 2001, Boston, MA, July 2001

[10] D. Trinchero, A. Galardini, R. Stefanelli, B. Fiorelli, P. Gianola, R. Scotti, and R. Vallauri, *"A Fraunhofer-based Approach for the Assessment of the Field Radiated in the Fresnel Region of an Antenna"*, LAWP 2008.

[11] D. Trinchero, R. Stefanelli, A. Galardini, P. Gianola, R. Scotti, R. Vallauri, *"Simplified Numerical Field Assessment in the Fresnel Region of the BTS Antennas"*, IEEE Int. Symposium on Antennas & Propagation and USNC/URSI National Radio Science Meeting, Charleston, SC, June 2009.

[12] D. Trinchero, A. Galardini, R. Stefanelli, P. Gianola, R. Scotti, R. Vallauri, *"A simplified and conservative method for electromagnetic field assessment in the Fresnel region of the radiator"*, 3rd European Conference on Antennas and Propagation (EuCAP 2009),Berlin, Germany, March 2009.

[13] A. Carta, R. Stefanelli, D. Trinchero, *"Approccio Semplificato ma Conservativo per la Valutazione del Campo Irradiato nella Regione di Fresnel"*, XVIII Riunione Nazionale di Elettromagnetismo (XVIII RiNEm), Benevento, Italia, September 2010.

[14] A. Carta, D. Trinchero, *"Generalized method for fast field assessment in the radiating near field region"*, IEEE International Symposium on Antennas and Propagation (APSURSI) 2011, July 2011.

[15] A. Carta, D. Trinchero, *"Analysis and improvement of a fast Fraunhofer method for antenna characterization in its radiating region"*, 2011 XXXth URSI General Assembly and Scientific Symposium, August 2011.

[16] A. Carta, D. Trinchero, *"Advanced simplified algorithm for electromagnetic field computation in the radiating near field of BTS antennas"*, 2011 IEEE International Symposium on Electromagnetic Compatibility (EMC), August 2011.

[17] "The international classification for Seasonal Snow on the Ground" Paris, 2009.

[18] C. Jaedicke, *"Snow Mass Quantification and Avalanche Victim Search by Ground Penetrating Radar"*, Surveys in Geophysics, 2003.

[19] J. Anderson, J. Wirt, *"Ultrasonic Snow Depth Sensor Accuracy, Reliability, and Performance"*, Western Snow Conference, Hood River, Oregon, 2008.

[20] J.A. Bergman, *"An Evaluation of the Acoustic Snow Depth Sensor in a Deep Sierra Nevada Snowpack"*, Proceedings of the 57th Annual Western Snow Conference, Fort Collins, CO, 1989.

[21] W.A. Brazenec, *"Evaluation of ultrasonic snow depth sensors for Automated Surface Observing Systems (ASOS)"*, M.S. thesis, Colorado State University, Fort Collins, CO, 2005.

[22] C. Calvo, J.L. Villarroel, J.A. CuchÃ, *"Valores de condcutividad elÃ©ctrica de nieve y suelo a lo largo de un transecto longitudinal en el Pirineo"*, Febraury 2006.

[23] N. Yankielun, W. Rosenthal, R. E. Davis, *"Alpine snow depth measurements from aerial FMCW radar"*, Cold Regions Science and Technology, 2004.

[24] J. Holmgren, M. Sturm, N. E. Yankielun, G. Koh, *"Extensive measurements of snow depth using FM-CW radar"*, Cold Regions Science and Technology, 1998.

[25] J. S. Deems, T. H. Painter, *"LIDAR measurements of snow depth: accuracy and error sources"*, International Snow Science Workshop, Telluride, Colorado, 2006.

[26] A. Ruiz Garcia, J. Marturia Alavedra, P. Oller Figueras, I. Moreno Banos, J. Pina Iglesias, A. Martinez Figueras, J. Talaya Lopez, *"Assessment of airborne LIDAR for snowpack depth modeling"*, Boletin de la Sociedad Geologica Mexicana, 2011.

[27] E. L. Peck, V. C. Bissell, *"Aerial Measurement of Snow Water Equivalent by Terrestrial Gamma Radiation Survey"*, Hydrological Sciences Bulletin, 1973.

[28] E. Paquet, M. Laval, L. M. Basalaev, *"An Application of Cosmic-Ray Neutron Measurements to the Determination of the Snow Water Equivalent"*, Proceedings of the 30th International Cosmic Ray Conference, Merida, Yucatan, Mexico, 2007.

[29] M. Kodama, K. Nakai, S. Kawasaki, M. Wada, *"An application of cosmic-ray neutron measurements to the determination of the snow-water equivalent"*, Journal of Hydrology, 1979.

[30] M. Kodama, *"Continuous monitoring of snow water equivalent using cosmic ray neutrons"*, Cold Regions Science and Technology, 1980.

[31] J. Stein, D. L. Kane, *"Monitoring the unfrozen water content of soil and snow using time domain reflectometry"*, Water Resources Research, 1983.

[32] A. Lundberg, *"Laboratory calibration of TDR-probes for snow wetness measurements"*, Cold Regions Science and Technology, 1997.

[33] M. Schneebeli, C. Coleou, F. Touvier, B. Lesaffre, *"Measurement of density and wetness in snow using time-domain reflectometry"*, Annals of Glaciology, 1998.

[34] P. Waldern, C. Huebner, M. Schneebeli, A. Brandelik, F. Rau, *"Continuous Measurements of Liquid Water Content and Density in Snow Usign TDR"*, Proceedings of the Second International Symposium and Workshop on Time Domain Reflectometry for Innovative Geotechnical Application, 2001.

[35] M. Stacheder, *"TDR and low-frequency measurements for continuous monitoring of moisture and density in a snow pack"*, Institute of Agrophysics, 2005.

[36] M. Tiuri, A. Sihvola, E. Nyfors, M. Hallikaiken, *"The complex dielectric constant of snow at microwave frequencies"*, IEEE Journal of Oceanic Engineering, December 1984.

[37] M. Hallikainen, F. Ulaby, M. Abdelrazik, *"Dielectric properties of snow in the 3 to 37 GHz range"*, IEEE Transactions on Antennas and Propagation, November 1986.

[38] Sihvola, Ari, Nyfors, Ebbe, Tiuri, Martti, *"Mixing formulae and experimental results for the dielectric constant of snow"*, Journal of Glaciology, 1985.

[39] C. Matzler, H. Aebischer, E. Schanda, *"Microwave dielectric properties of surface snow"*, IEEE Journal of Oceanic Engineering, December 1984.

[40] M. Stacheder, F. Koeniger, R. Schuhmann, *"New Dielectric Sensors and Sensing Techniques for Soil and Snow Moisture Measurements"*, Sensors, 2009.

[41] A. Denoth, I. Wilhelmy, *"Snow Dielectric Devices and Field Applications"*, International Snow Science Workshop, Whistler, 1988.

[42] A. Sihvola, M. Tiuri, *"Snow Fork for Field Determination of the Density and Wetness Profiles of a Snow Pack"*, Geoscience and Remote Sensing, 1986.

[43] J.R. Kendra, F.T. Ulaby, K. Sarabandi, *"Snow Probe for In Situ Determination of Wetness and Density"*, Geoscience and Remote Sensing, 1994.

[44] A. Kavak, W.J. Vogel, G. Xu, *"Using GPS to Measure Ground Complex Permittivity"*, Electronics Letters, 1998.

[45] V.U. Zavorotny, A.G. Voronovich, *"Bistatic GPS Signal Reflections at Various Polarizations from Rough Land Surface with Moisture Content"*, Geoscience and Remote Sensing Symposium, 2000.

[46] S.T. Lowe, P. Kroger, G. Franklin, J.L. LaBrecque, *"A Delay/Doppler-Mapping Receiver System for GPS-Reflection Remote Sensing"*, Geoscience and Remote Sensing, 2002.

[47] D. Masters, P. Axelrad, S. Katzberg, *"Initial results of land-reflected GPS bistatic radar measurements in SMEX02"*, Remote Sensing of Environment, 2004.

[48] S. J. Katzberg, O. Torres, M. S. Grant, D. Masters, *"Utilizing calibrated GPS reflected signals to estimate soil reflectivity and dielectric constant: Results from SMEX02"*, Remote Sensing of Environment, 2005.

[49] A. Della Vecchia, P. Ferrazzoli, L. Guerriero, I. Cacucci, M. Marzano, *"Optimization of bistatic radar configurations for vegetation monitoring"*, Geoscience and Remote Sensing Symposium, 2006.

[50] M. S. Grant, S.T. Acton, S. J. Katzberg, *"Terrain Moisture Classification Using GPS Surface-Reflected Signals"*, Geoscience and Remote Sensing Letters, 2007.

[51] T. Lindgren, "Algorithm Development for Multistatic GNSS Radar Technology", M.S. Thesis, Lulea University of Technology, 2007.

[52] O. Nogues-Correig, E. Cardellach Gali, J. Sanz Campderros, A. Rius,*"A GPS-Reflections Receiver That Computes Doppler/Delay Maps in Real Time"*, Geoscience and Remote Sensing, 2007.

[53] N. Pierdicca, L. Pulvirenti, L. Guerriero, G. Della Pietra, "Optimal configurations of bistatic radar for retrieving soil moisture and vegetation biomass" , Geoscience and Remote Sensing Symposium, 2007.

[54] M. Kebiao, Z. Mengyang , W. Jianming, T. Huajun, Z. Qingbo, *"The Study of Soil Moisture Retrieval Algorithm from GNSS-R"*, International Wokshop on Geoscience and Remote Sensing, 2008.

[55] S. Jin, A. Komjathy, *"GNSS Reflectometry and Remote Sensing: New Objectives and Results"*, Advances in Space Research, 2010.

[56] V.U. Zavorotny, K.M. Larson, J.J. Braun, E.E. Small, E.D. Gutmann, A.L. Bilich, *"A Physical Model for GPS Multipath Caused by Land Reflections: Toward Bare Soil Moisture Retrievals"*, Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2010.

[57] S. Jin, G.P. Feng, S. Gleason, *"Remote sensing using GNSS signals: Current status and future directions"*, Advances in Space Research, 2011.

[58] M. Konig, J.G. Winther, E. Isaksson, *"Measuring snow and glacier ice properties from satellite"*, Reviews of Geophysics, 2001.

[59] M.D. Jacobson, *"Inferring Snow Water Equivalent for a Snow-Covered Ground Reflector Using GPS Multipath Signals"*, Remote Sensing, 2010.

[60] M.D. Jacobson, *"Snow-covered lake ice in GPS multipath reception-Theory and measurement"*, Advances in Space Research, 2009.

[61] K. M. Larson, E. D. Gutmann, V. U. Zavorotny, J.J. Braun, M.W. Williams, F. G. Nievinski, *"Can we measure snow depth with GPS receivers?"*, Geophysical Research Letters, 2009.

[62] S. Gleason, *"Towards Sea Ice Remote Sensing with Space Detected GPS Signals: Demonstration of Technical Feasibility and Initial Consistency Check Using Low Resolution Sea Ice Information"*, Remote Sensing, 2010.

[63] K.M. Larson, F.G. Nievinski, *"GPS Snow Sensing: Results from the Earth Scope Plate Boundary Observatory"*, GPS Solutions, 2012.

[64] J. Stepanek, D. W. Claypool, *"GPS signal reception under snow cover: A pilot study establishing the potential usefulness of GPS in avalanche search and rescue operations"*, Wilderness & Environmental Medicine, 1997.

[65] H. Kuusniemi, G. Lachapelle, J. H. Takala, *"Position and velocity reliability testing in degraded GPS signal environments"*, GPS solutions, 2004.

[66] J.B. Schleppe, G. Lachapelle, *"GPS Tracking Performance Under Avalanche Deposited Snow"*, Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS), 2006.

[67] A. Wieser, *"High-sensitivity GNSS: The trade-off between availability and accuracy"*, 3rd IAG - 12th FIG Symposium, Baden, 2006.

[68] M. Petovello, G. Lachapelle, *"ELoran and Signal Reception under snow"*, GPS Solutions, 2007.

[69] J. B. Schleppe , G. Lachapelle, *"Tracking performance of a HSGPS receiver under avalanche deposited snow"*, GPS Solutions, 2008.

[70] S. Colbeck, *"An Overview of Seasonal Snow Metamorphism"*, Rev.Geophysics, 1982.

[71] F. Techel, C. Pielmeier, *"Point observations of liquid water content in wet snow investigating methodical, spatial and temporal aspects"*, The Cryosphere, 2011.

[72] C. R. L. Fierz, Y. Durand, P. Etchevers, , E. Green, D. McClung, K. Nishimura, P. Satyawali, S. Sokratov, *"The International Classification for Seasonal Snow on the Ground"*, IHP-VII Technical Documents in Hydrology N 83, IACS Contribution N 1, UNESCO-IHP, Paris, 2009.

[73] E. B. Jones, A. Rango, S. M. Howell, *"Snowpack liquid water determinations using freezing calorimetry"*, Nordic Hydro, 1983.

[74] A. Carta, R. Stefanelli, D. Trinchero, *"Microwave Electromagnetic Sensors to Estimate Water Density Within Snow"*, XIX Riunione Nazionale di Elettromagnetismo (XIX RiNEm), Roma, Italia, September 2012.

[75] A. Carta, A. Ghaghazanian, R. Stefanelli, D. Trinchero, *"Radiofrequency sensors for snow conditions monitoring and real time avalanche alerts"*, 2013 IEEE Radio and Wireless Week (RWW 2013), January 2013.