APET: Antenna Pattern Extraction Tool
An open source software to discover real-world HF-antenna behavior

Introduction

Recently I built a new antenna, a sort of Lazy H on a single aluminum pole with the elements bent inwards and connected to guying lines (its model and 3D pattern are visible in Fig. 1). We all know what happens just after finishing this kind of work: we immediately go to our shack and try to make as many QSOs as possible, maybe asking for comparisons between the new and a known antenna.

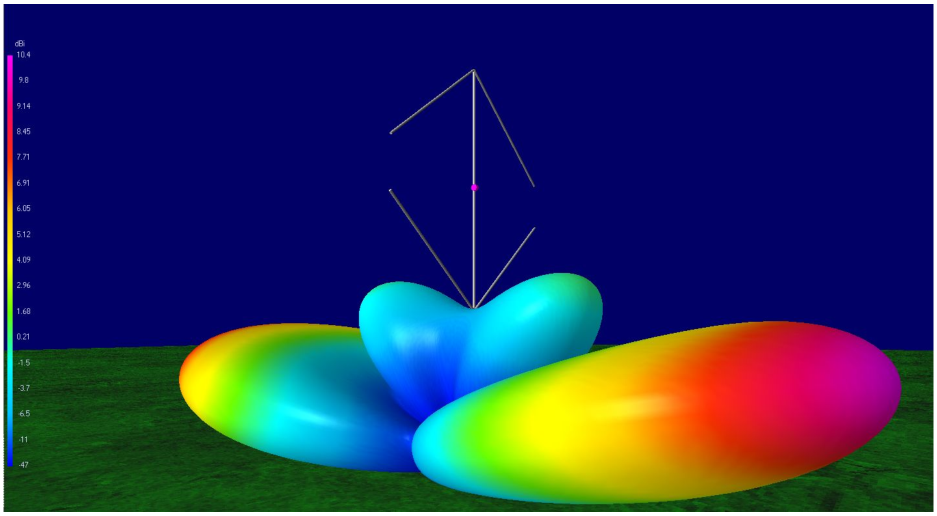

Fig. 1: The recently built antenna shows a promising theoretical 3D radiation pattern, but will it be confirmed by experimental evidence? *(Antenna modeled in MMANA)*

This time I tried to be more systematic and, since there was a huge worldwide contest on 20m, I began to tune on as many stations as possible and wrote down the signal difference between antennas. I am lucky enough to live in a rural area with very low noise and I have a networked kiwiSDR connected to a wideband LZ1AQ magnetic loop. This loop is quasi omnidirectional on 14 MHz for elevation angles above 1-2 degrees, so I decided it was a good reference antenna for my Lazy H initial characterization. So I spent a few hours tuning around, estimating signals with a calibrated S-meter and writing down a long list from all possible directions. I ended up with a nice table that my wife helped me to convert into a spreadsheet. The final plot was quite nice even if I was not too sure about some signal values of stations I heard just for a few seconds and then disappeared.

Next day at work I showed this rough horizontal antenna pattern plot to a non-ham friend of mine (he's an engineer, I'm a physicist) and he immediately said: “Nice plot! Of course you did this automatically! It must be a real pain to do it by hand…”. At that point I realized that I needed automatic beacons on all bands, on at all times and a way of measuring the incoming signal in a reliable way. The obvious answer was: “WSPR!”. Each WSPR spot contains date, time, callsign, power, SNR, locator and a few more parameters.

So the idea began to take shape by setting up in my shack two independent receiver chains: two antennas, two receivers, two PCs to decode and upload spots to WSPR database. I decided to “decorate” one of my callsigns with a final number in order to be able to later distinguish which RX received who. Nice, so from writing down hundreds of stations on paper, we're just writing down at what time we start our session.

The data now resides on a remote database that everyone may access: each experiment is saved “forever” and I just need to remember the date, time and how I decorated the callsigns.

One might argue that precise methods to extract the antenna radiation pattern already exist, for instance by means of a small mobile transmitter to be driven around and someone who maps the received signal, and of course there is expensive lab equipment for smaller antennas at UHF frequencies. But for HF it's not that practical to have a friend driving around with a transmitter and us writing down signal levels by hand… One could maybe use a rotator, just one faraway beacon, but signals would come just from one location/distance and the propagation oscillations would badly ruin our data in case of fading. The main advantage of the method presented here is that we can measure the antenna in its environment *and* with the real long-distance signals from all over the world that it will have to deal with when in real operation, not just nearby beacons. The ground reflection of a low angle signal, that greatly contributes to our directional antenna vertical pattern, may happen even a few kilometers away from the antenna itself. No numerical simulation software exists to model such a complex and vast landscape, so it would be extremely useful, for example, to be able to measure, not just guess, at which height a Yagi should be mounted to obtain the best performance, if we live on a hill in front of the sea.

Moreover, this method should prove very useful to combat noise, helping to choose the quietest antenna not just using anecdotal evidence like "mag loops are quieter than dipoles" or "verticals are noisy" [1]. Another context would be evaluation of balun efficiency (or just to see if one is needed) to avoid pattern deformation due to current unbalance, that usually translates to: common mode reception noise (your cable is acting like a part of the antenna) and unwanted lobes or insufficient front/side or front/rear gain ratios. Finally, being developed as a reception only approach, we don't need to increase the pollution in the WSPR narrow bands and, more importantly, the method is applicable to receive antennas for the low and very low bands in which absolute gain is not that important, but directionality factor really is[2].

Method and some results

What we need to automate our process is to obtain a long list of spots (don't overdo, we'll see later why), each decoded by two independent receivers/antennas. We'll discard all spots that were received by just one RX since we're looking for signal differences, not just signals. The WSPR algorithm computes not a signal level, but rather a Signal to Noise Ratio (SNR) in dB that is much more useful, because a receive antenna could have a much higher signal than another, but a much higher noise background. So we're now in a position to take *differences of SNRs*:

$$\Delta SNR = SNR_1 - SNR_2 = (S_1 - S^1{}_N) - (S_2 - S^2{}_N) = \Delta S - \Delta N$$

[1] See for example https://www.w8ji.com/noise.htm for a useful discussion about noise.
[2] See https://www.w8ji.com/receiving.htm for a nice explanation of receive antenna evaluation.

If we assume that the noise in our location is sufficiently constant and we're able to measure it with both receivers, we can measure ΔN, thus the difference of signal intensities from the SNRs. Please note that this step is only important if you want to obtain "absolute" radiation patterns: if you just want to understand how much an antenna is better than another at some direction (or time of the day), you're fine with comparing SNRs. At my place, when the sun is quiet, the two antennas show roughly the same low noise so I just ignore it. In urban situations, where several noise sources from all directions exist, the direct use of SNRs would produce a quite different result from the standard antenna gain pattern: the *antenna SNR pattern*. This pattern depends strongly on how each antenna couples to nearby signals depending on polarization, distance, and many other factors. Our method may also indirectly show a common mode noise ingress since one antenna would have a totally different pattern from the one computed numerically.

Of course the whole idea expressed so far could be reversed and we could set up two TX stations in the same location and collect the spots uploaded by the other receivers. This configurations looks very similar in principle to the original one, but it has a fundamental difference: we have no control on the noise at the receiving side, antenna pattern nor sensitivity. On the other hand, we have, hopefully, full control over our own receivers/antennas.[3] Each method is useful since an antenna may be good for receiving purposes and totally useless at transmitting and vice versa. The software is already capable of handling both scenarios with minimal changes.

Back to our original idea: With our long list of common spots and the associated (pair of) SNR values we can easily plot the time evolution of each received station: remember, we can do it both for single SNRs and for the derived ΔSNR. Before we proceed any further, we should take a small step back, since we assumed our twin RX chains were ideal: our two receivers could have a vastly different Noise Figure, some cable could be bad, or maybe different versions of the WSPR software (*don't do that! WSPR 2.0 is ~3dB more sensitive than 1.9 for instance.*). So what we should do before getting serious about plotting meaningful data is to connect the same antenna via a hybrid splitter to both receivers and decode (concurrently) a few dozens spots. This allows us to check if they're equal in number and in

[3] At the moment of writing this article, I received a message in a online forum informing me that a similar WSPR transmission approach exists: https://www.sotabeams.co.uk/wsprlite-classic

SNR: if there's a systematic variation in SNR levels for the same spots we should take that into account. If this step produces very different results, you have to solve it before proceeding further. Personally, I'm using a kiwiSDR[4] as the first receiver and a Kenwood TS-940S as secondary and I measured a consistent offset of 3 dB between SNRs in favor of the latter.[5]

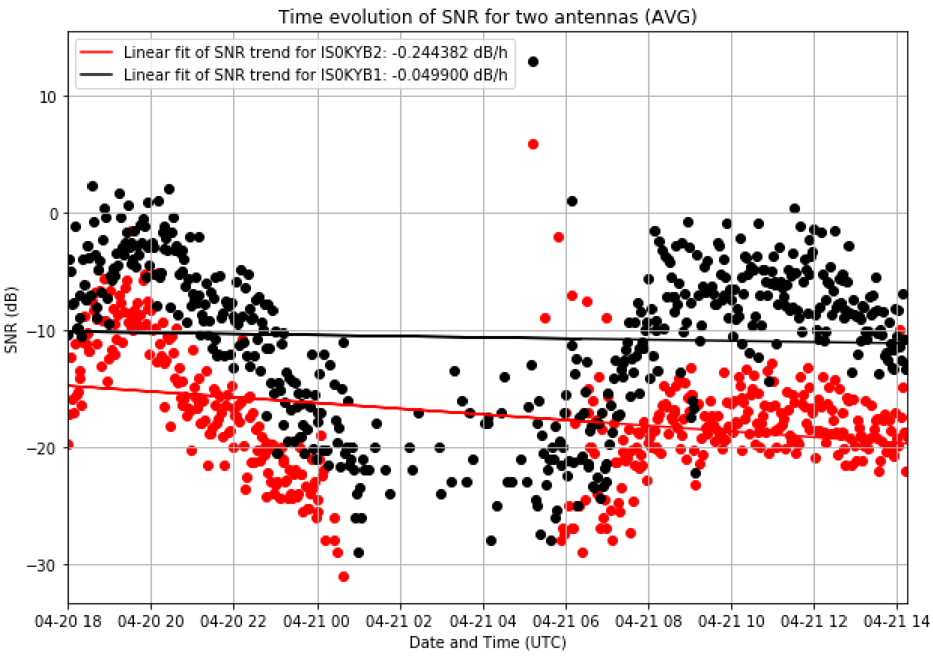

Fig. 2: Example of time evolution of the WSPR SNR values obtained from both antennas. The difference between the two is far from constant and the loop totally stops receiving at night. The variable electron density distribution in the ionosphere plays a major role in establishing the angle of arrival of each signal.

After reconnecting each antenna to its receiver, we're ready to observe the evolution of the band over some time (see Fig. 2). What we're looking for are periods of activity in which both RXs produce many common spots: if one of your antennas is totally deaf, remember, you're not producing useful statistics: you'll just be able to plot how one receiver performs

[4] Try to avoid using the integrated kiwiSDR WSPR decoder and follow http://ka7oei.blogspot.com/2018/10/a-quick-and-probably-incomplete-guide.html it's much more sensitive!

[5] A Microtelecom Perseus in place of the Kenwood produced SNR values offset by 1dB with respect to the kiwiSDR.

while the other is dead. Try to avoid periods when the SNR evolution has a steep slope: that is usually caused by changing ionospheric conditions. In Fig. 2, one would usually choose the period between 8:00 and 14:00 UTC, but results are not too sensitive and one should only worry if performing accurate measurements. All antenna patterns shown in the rest of the article are produced in that time window even if during that time of the day very few spots from the Americas were received.

The whole idea of comparing antennas with different polarizations would be hopeless if there wasn't the ionosphere to randomly rotate signal polarization. Moreover, if our SNR samplings were very fast, each antenna could receive a very different signal due to polarization effects. We're able to avoid this phenomenon thanks to the time averaging (2 minutes) of each WSPR transmission: the SNR is computed over a time interval in which roughly one full polarization rotation takes place for the same spot. M.R. Epstein (1969), affirms: *"Daytime rates of polarization rotation with time (at a given frequency) average 0.25 turn/min. The polarization rotation rates with time do not appear to vary either with path azimuth or transmitted radio frequency. Near-zero rates of polarization rotation with time occur for much of the nighttime period"*.[6] You may still consider two minutes not enough to average these effects, that's why we use, for each received station, the median value of the ΔSNR during long sequences of spots. The standard deviation of this quantity and the length of the sequence are useful as reliability indicators and to be used as weights for the pattern smoothing and interpolation (Fig. 5).

[6] *Polarization of Ionospherically Propagated HF Radio Waves with Applications to Radio Communication,* M. R. Epstein, January 1969, Radio Science.

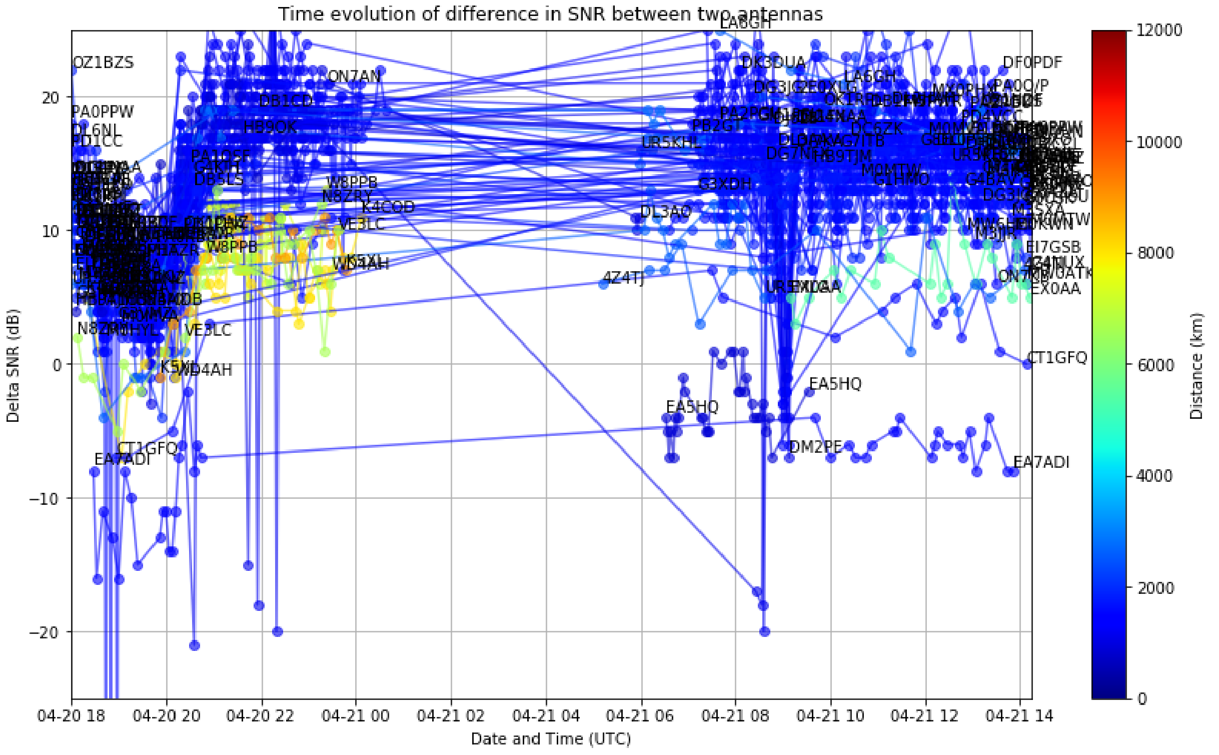

Fig. 3: How the ΔSNR evolves with time over about 20 hours. Each line of consecutive spots of the same station is colored depending on distance. Blue lines are mostly european stations, green, yellow and orange are mostly american ones. The large gap in the middle is due to one antenna not receiving any spot during the night. The gain of the directional antenna may change of 10-12 dB for the same station in a few hours.

In Fig. 3 we can clearly distinguish several propagation changes:

1. first day, between 8:00 and 19:30 UTC, the directional antenna vs loop advantage shrinks very rapidly for EU stations (blue lines) as the F2 layer thins and reaches higher altitudes, so producing signals from very low angle of arrival;
2. until 21:00 UTC the ionosphere stabilizes into its nocturnal state and american stations (green, yellow, orange) improve their signals on the directional antenna;
3. between 22:00 and 6:00 UTC of the next day no signals are spotted on the loop antenna (as seen on Fig. 2: no red dots) so no data can be used;
4. during the last day signals from Northern Europe are much stronger on the directional antenna, on average between 10-20 dB, and conditions remain stable until 14:00 UTC.

Conclusion

You may be wondering now, how the antenna I talked about at the beginning actually performs in real life: it seems to work very well and it surpassed my expectations, but I never expected to see the wild variations of gain with respect to my loop antenna along the day. Moreover, I was expecting to cover the horizon with two such bidirectional Lazy-H but this software showed that even on 20m the radiation lobes are quite narrow and I should think of making it rotatable. This conclusion would probably have remained at the anecdotal level if I never built the software…

With the software in its current state, you may explore several useful statistics:

1. SNR levels over time (Fig. 2) for both RX chains: is one RX “getting better” over time than the other?;
2. ΔSNR between the two RXs over time (Fig. 3);
3. How the probability distribution of spot distances evolve: is the skip getting longer, etc: so basically, we may acquire statistics on which antenna to use at specific times/bearings (Fig. 3);
4. Filter the database by callsign, by top spotted stations, by distance, by locator with some radius, by continent, by bearing, or whatever comes to your mind;
5. Get the polar distribution of SNR (or signal) differences (Fig. 4);
6. Interpolate the data and exploit intrinsic antenna symmetries to get an accurate version of your azimuthal plot (Fig. 5);
7. Superimpose your experimental data to your theoretical model patterns for both antennas and their difference (Fig. 6).

You must bear in mind that the software presented here is currently evolving at a fast rate and you should not expect a polished interface (there is none!) or ease of use with nice buttons or “instructions”… You should instead expect a heavily commented Python code that you may use interactively even if you’re not a programmer (but it helps if you know Python basics and how to use a generic NEC antenna simulator). The project is released as open source under the GPL V.3 licence to encourage collaboration and sharing of ideas.

You may freely download the software from GitHub: https://github.com/mcogoni/APET and execute on your computer or directly on the cloud.

This article doesn't want to sell you a complete product to analyse antennas and propagation, but its main goal is to present a unified framework to be used as it is by amateurs with simple objectives, but hopefully to be easily extended by advanced experimenters who may want to:

- achieve better results with their antennas;
- better understand how propagation influences the differences between antennas;
- collect statistics on various types of propagation all year long;
- simply go beyond claims such as "a Yagi is always X dB over a dipole" or "verticals are noisy".

Future work will probably try to exploit real-time global ionospheric data to compute the probability distribution of the angle of arrival for each signal. This would enable us to go beyond a 2D azimuthal pattern by acquiring a full 3D (even if rough and probabilistic) antenna pattern. Extension of these ideas to more than two antennas would open up research to even more advanced topics as taking into account polarization rotations (and not just averaging over them). Another nice idea would be to compare the reception of nearby, or even not so distant, receiving stations: this would allow us to compute geographic correlations and much more.

Acknowledgments

I should thank my colleagues and friends Giovanni Busonera and Massimo Gaggero (IS0FEO), and my remote radio mentor Leif Åsbrink (SM5BSZ) for useful discussions on several topics presented here.

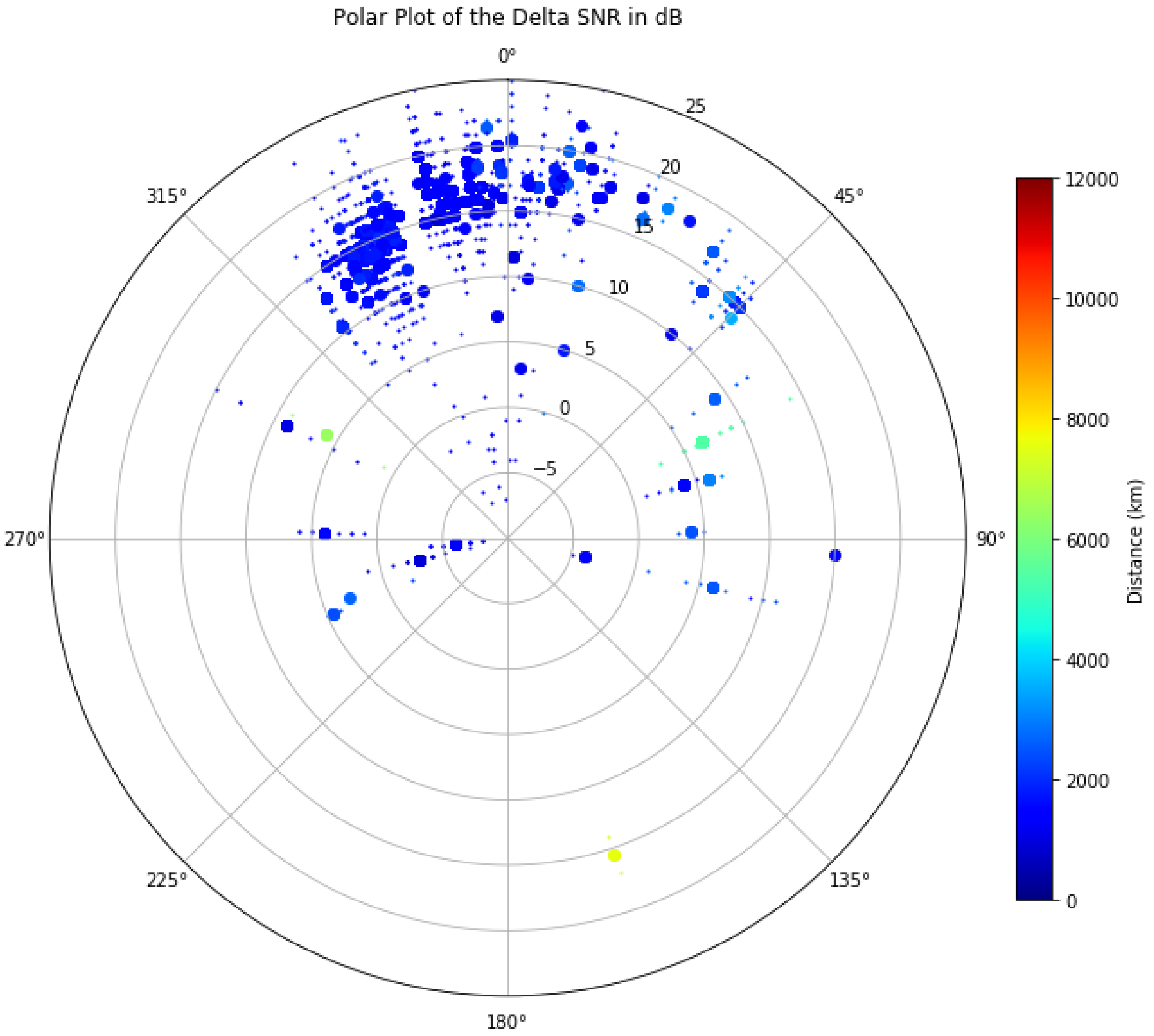

Fig. 4: Polar distribution of ΔSNR over ~20 hours: each circle represents the ΔSNR in dB. Color represents distance as in Fig. 3. Big dots are the median value in dB over the whole set of spots for each station, whereas small dots are single spots.

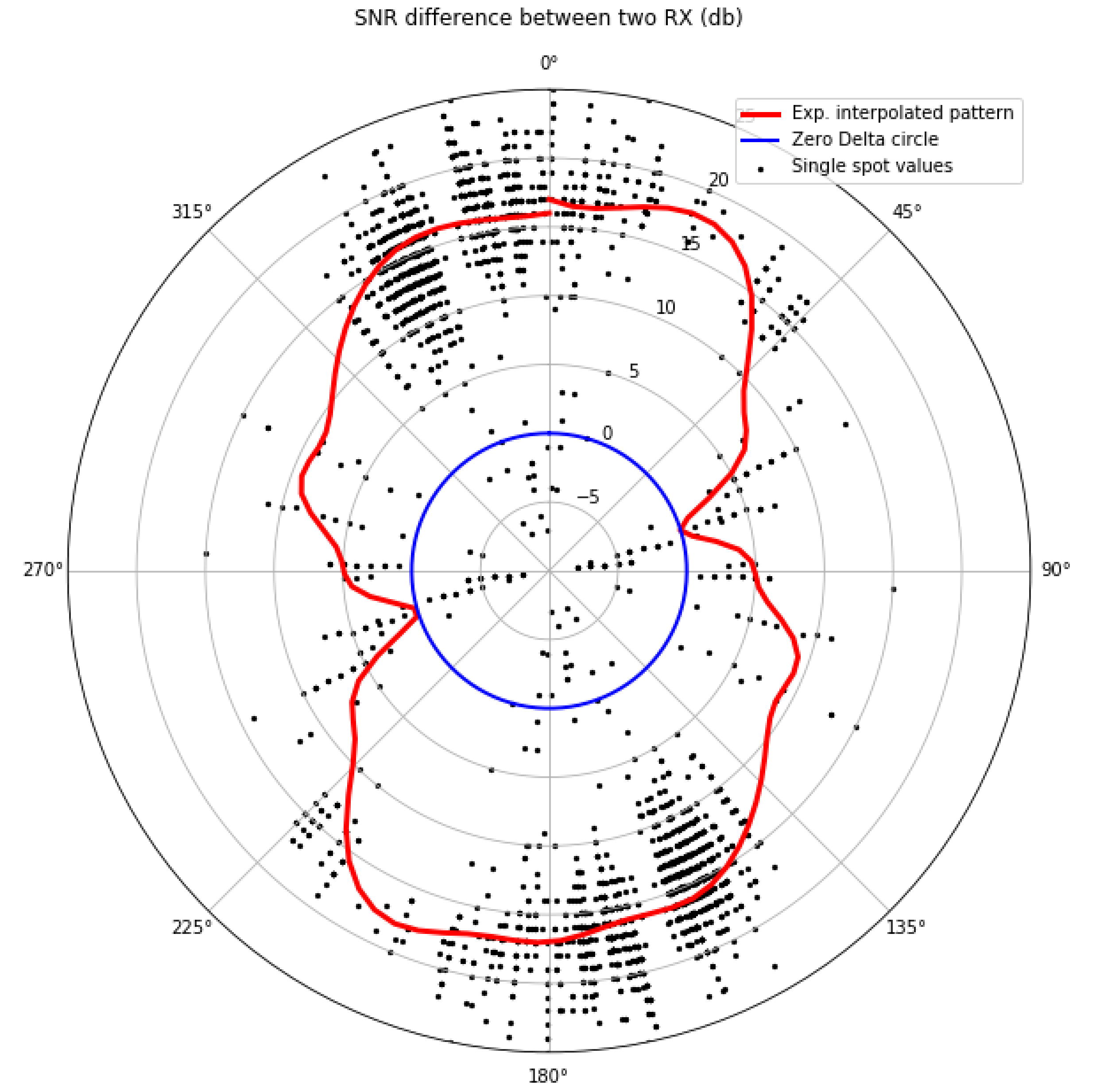

Fig. 5: Cubic spline interpolation of the ΔSNR: here we exploited the axial symmetry of the two antennas to artificially improve the statistics, especially poor from the South direction. The interpolation is performed over the selected 6 hours of data and each black dot is a single value for every spot of all stations.

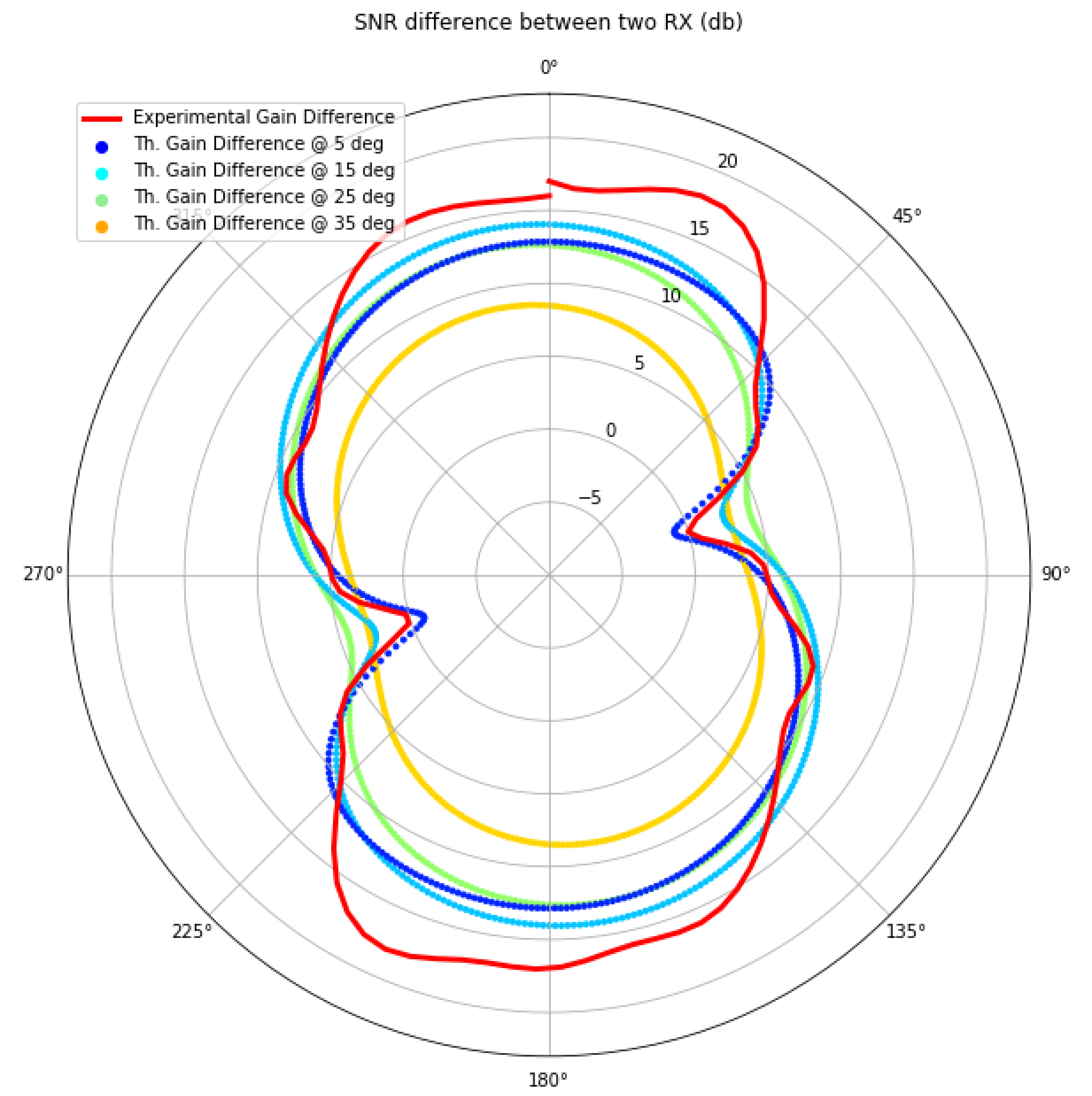

Fig. 6: Comparison of the “antenna difference pattern” obtained so far with respect to the difference of the theoretical NEC models at four elevation angles: 5, 15, 25 and 35 degrees. The resulting pattern is mostly within 3 dB of the predicted one.