Time transfer and Global Navigation Satellite Systems (GNSS) spoofing, & spoofing detection

G. Goavec-Merou[1], J.-M Friedt[1], F. Meyer[2]

[1] FEMTO-ST/temps-fréquence & FAST-LAB, Besançon
[2] OSU Théta/Observatoire de Besançon & FAST-LAB, Besançon

jmfriedt@femto-st.fr

slides at `jmfriedt.free.fr/fosdem2019_gps.pdf`
presentation at
`https://video.fosdem.org/2019/AW1.120/sdr_gps.mp4`

sequel to "Software Defined Radio for processing GNSS signals (FOSDEM 2015)"

GPS

1. NAVSTAR: military program started in 1973 (sats launched in 1978)
2. Clinton cancels Selective Availability in May 2000, dropping the resolution from $\simeq$ 45 m to $\simeq$ 5 m [1]
3. Positioning as a result of trilateration of space-borne atomic clock-synchronized signals
4. Growing access to Software Defined Radio (SDR) for receiving and *synthesizing* the signals
5. Spoofing GPS has become a sub-100 euro activity: what consequences ?
6. Computationnally efficient spoofing detection using antenna array

Figure: US Air Force

[1] www.gps.gov/systems/gps/modernization/sa/data/

GPS

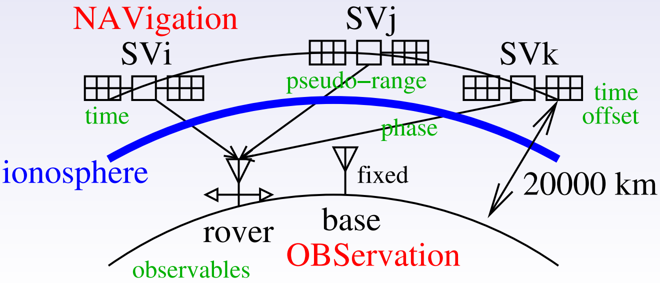
1. NAVSTAR: military program started in 1973 (sats launched in 1978)
2. Clinton cancels Selective Availability in May 2000, dropping the resolution from $\simeq$ 45 m to $\simeq$ 5 m
3. Positioning as a result of trilateration of space-borne atomic clock-synchronized signals
4. Growing access to Software Defined Radio (SDR) for receiving and *synthesizing* the signals
5. Spoofing GPS has become a sub-100 euro activity: what consequences ?

The importance of technical advances in measuring time was underscored by European regulations that went into effect in January and that require financial institutions to **synchronize time-stamped trades with microsecond accuracy**. *Being able to trade* **at the nanosecond level** *is vital to Nasdaq. Two years ago, it debuted the Nasdaq Financial Framework, a software system that it has envisioned eventually trading everything from stocks and bonds to fish and car-sharing rides.*
[...]
Google would later use this method to synchronize computers **based on GPS data** *and atomic clocks to make sure that their database system could correctly order transactions. But since the system requires super-accurate clocks and satellite receivers, it is more costly than the software-based Huygens approach.*

"Time Split to the Nanosecond Is Precisely What Wall Street Wants"
The New York Times (John Markoff, June 29, 2018)

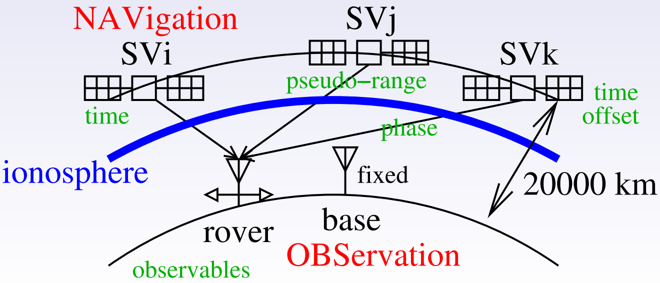
GNSS basics: space & ground segments

- Position = trilateration of timing signals emitted from space
- Pseudo-range=distance from receiver to each satellite ⇒ trilateration for position identification
- Spaceborne atomic clock offsets measured with respect to ground atomic clocks: delay information in NAVigation message
- electromagnetic waves : 300 m/μs ⇒ 3 m accuracy requires **10 ns accuracy**
- High accuracy position = **precise time transfer**



GNSS basics: space & ground segments

- Navigation data represent the constellation, observations are collected by the ground based receiver
- Standardized data format: RINEX Receiver INdependent EXchange
- RINEX ephemeris are published for improved accuracy of receiver position (better satellite position measurement than prediction, ionospheric delay) with an hourly delay
- raw ground based measurements: **pseudo-range** is the uncorrected measurements from satellite to ground station



GNSS basics: timing information

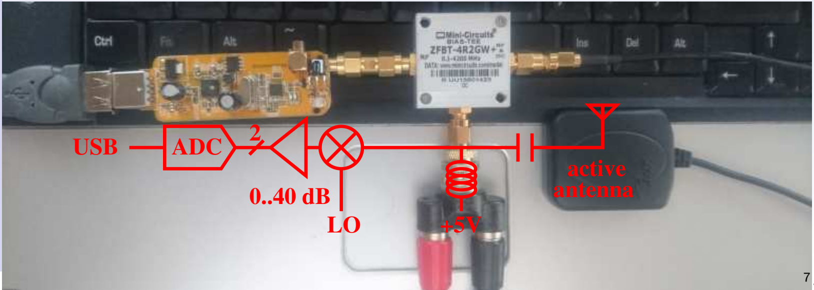
- Common view time transfer: observe local clock offset wrt GNSS, and inform of this offset (UTC) – AF0, AF1 & AF2 fields [1]
- GPS provides a precise time reference (GPS time) materialized through the 1-PPS pulse (rising edge is exact on the GPS second within $\pm$100 ns)
- The GPS sends a "standard sentence" (NMEA or 1-PPS information) *prior* to the pulse to indicate the date and time. [2]

[1] ESA TM-23 guidebook on GNSS processing: www.navipedia.net/index.php/GNSS:Tools: ESA books on GNSS processing www.navipedia.net/GNSS_Book/ESA_GNSS-Book_TM-23_Vol_I.pdf & www.navipedia.net/GNSS_Book/ESA_GNSS-Book_TM-23_Vol_II.pdf

[2] https://www.trimble.com/ec_receiverhelp/v4.15/en/ioConfig.html#1PPS: "1PPS Time Tag – Enables the ASCII Time tags. The time tag provides the UTC time of the 1PPS pulse and is output approximately 800 milliseconds before the pulse."

GNSS basics: Doppler shift & link budget

- GNSS = classically MEO (Medium Earth Orbit) at 20000 km
- Standards defines the *received* power from which the emitted power can be deduced
- 50 W output power and link budget ⇒ received power below thermal noise
- Correlation with known pseudo-random pattern (Gold codes) for pulse compression (30 dB)
- Moving satellite: celestial mechanics defines the possible Doppler shift (±5 kHz)



CDMA: software decoding of GPS

- GPS: 31-satellite fleet [3] orbiting Earth at a distance of 20000 km
- Time reference (Cs+Rb and then Rb only)
- Time of flight computation for positioning
- Offsets introduced by electromagnetic wave velocity fluctuations (ionosphere, troposphere) impossible to compensate for if a single frequency carrier is monitored
- Satellite ephemeris + time of flight = position of receiver on Earth
- Multiple applications beyond positioning [4] [5]

All satellites transmit on the same carrier frequency

---

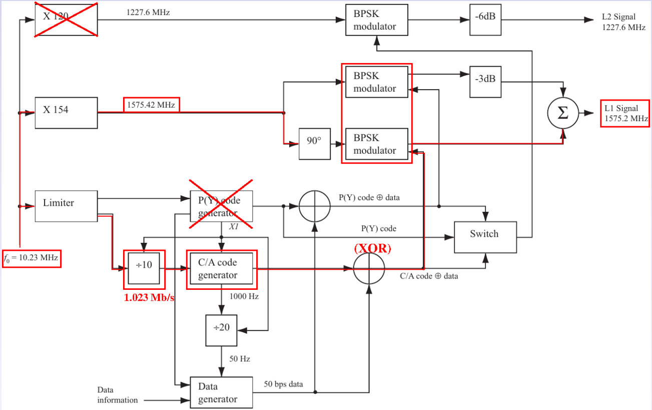
[3] `http://spaceflightnow.com/2014/10/13/gps-modernization-continues-with-quick-pace-of-launches/`

[4] J.-M Friedt, G. Cabodevila, *Exploitation de signaux des satellites GPS reçus par récepteur de télévision numérique terrestre DVB-T*, OpenSilicium 15, Juil.-Sept. 2015

[5] L. Lestarquit *et al.*, *Reflectometry With an Open-Source Software GNSS Receiver: Use Case With Carrier Phase Altimetry*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **9** (10), pp. 4843–4853 (2016)

CDMA: decoding GPS

GPS signal encoding principle [6] :

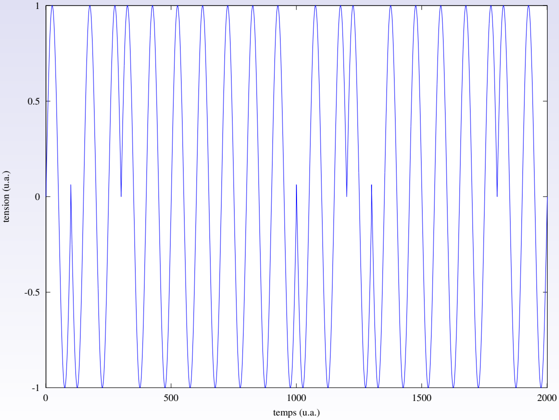


- the carrier is generated by an atomic clock (1575.42 MHz) ...
- ... is phase modulated at 1.023 MHz with a unique satellite identifier ...
- ... and again phase-modulated with the navigation message (50 bps)

[6] K. Borre *et al.*, *A Software-Defined GPS and Galileo Receiver – A Single-Frequency Approach*, Birkhäuser Boston, 2007

Phase modulation

- PSK : Phase Shift Keying
- $\varphi = \arctan(Q/I)$: output of the I/Q demodulator
- local oscillator stability – constellation diagram
- GPS: BPSK (Binary Phase Shift Keying) – demonstration using a saturated mixer controlled by the bits to be transmitted

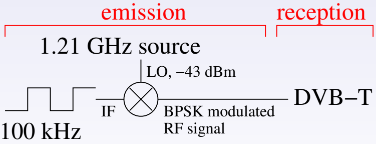


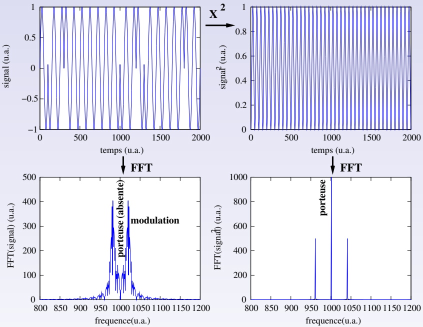
Phase demodulation

Receiver local oscillator is not synchronized on emitter oscillator

Phase recovery requires accurate reproduction of the unmodulated carrier

$$\exp(j(\Delta\omega t+\varphi))^N = \exp(j(N\Delta\omega t+N\varphi)) = \exp(jN\Delta\omega t) \text{ if } \varphi = 2\pi\cdot n/N$$





$\cos(\varphi)^2 \propto \cos(2\varphi)$

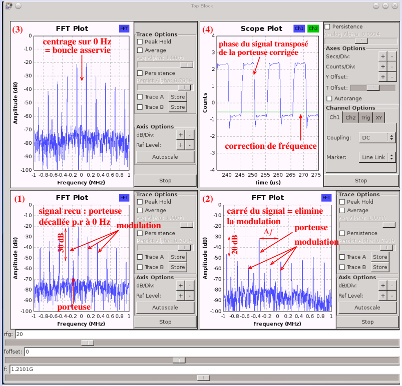
$\varphi \in [0;\pi] \Rightarrow 2\varphi = 0[2\pi]$

Carrier recovery by **squaring** BPSK

Phase demodulation

Transmitted signal is $s(t) = \exp(j(\underbrace{\omega_{TX}}_{\text{TX LO}} t + \underbrace{\varphi(t)}_{info}))$ so received signal is

$$s(t) = \exp(j(\omega_{TX}t+\underbrace{\varphi(t)}_{info}))\cdot\exp(-j\underbrace{\omega_{RX}}_{\text{RX LO}}t) = \exp(j(\underbrace{(\omega_{TX}-\omega_{RX})}_{\delta\omega}t+\varphi(t)))$$



Phase is time-integral of frequency

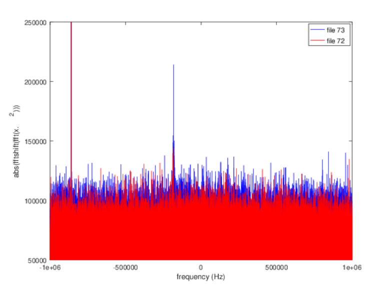
$\Phi = \delta\omega t + \varphi(t)$: we must have $\delta\omega = 0$ to recover $\varphi \in \{0; \pi\}$

Trick: $s^2(t) = \exp(j(2\delta\omega+2\varphi))$ and $2\varphi \in \{0; 2\pi\} = 0[2\pi]$
$\Rightarrow s^2(t)$ is a pure tone at $2\delta\omega$

Use the ready made Costas loop block

Example of GPS (BPSK)

Squaring a BPSK signal gets rid of modulation and collects all the energy in the carrier (requires averaging multiple Fourier transforms to get the squared signal spectrum out of the noise)

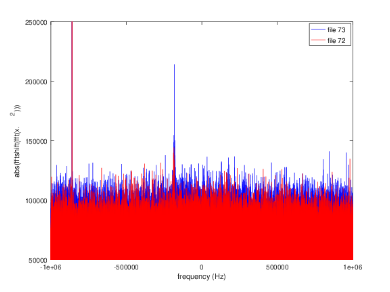


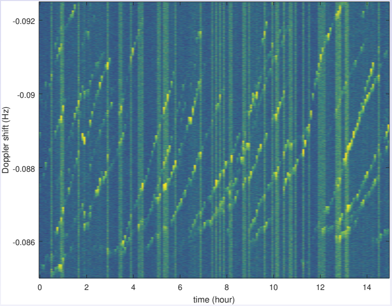
Coarse estimate of (twice) the Doppler shift+frequency offset[7]: "codeless" tracking in which each satellite is identified by its Doppler shift

[7] P. Boven, *Observe, Hack, Make: GPS* (2013): used in Vaisala RS80 radiosonde

Example of GPS (BPSK)

Squaring a BPSK signal gets rid of modulation and collects all the energy in the carrier (requires averaging multiple Fourier transforms to get the squared signal spectrum out of the noise)





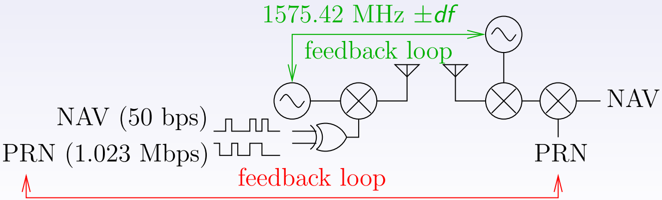
Coarse estimate of (twice) the Doppler shift+frequency offset[7]: "codeless" tracking in which each satellite is identified by its Doppler shift

[7] P. Boven, *Observe, Hack, Make: GPS* (2013): used in Vaisala RS80 radiosonde

Objectives

- a modulator generates the information, here encoded in the **phase of the carrier**
- the information is carried on a signal whose frequency varies (Doppler, thermal drift of LO): **phase is the integral of frequency**
- recovering the transmitted information is a matter of eliminating carrier information (requires a local copy)
- two degrees of freedom (carrier frequency and CDMA for satellite identification) will require two feedback loops to recover the information

⇒ carrier recovery and code position (delay) recovery



CDMA: decoding GPS

Cross-correlation: search for a (known) pattern $p(t)$ in the received signal $s(t)$.

$$xcorr(\tau) = \int_{-\infty}^{+\infty} s(t) \times p(t+\tau)dt$$

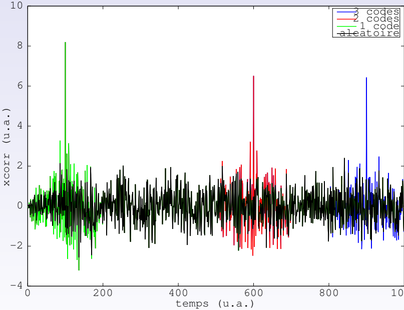
becoming for discrete time

$$xcorr(n) = \sum_{k=-\infty}^{+\infty} s(k) \times p(k+n)$$

Searching for a known pattern in an apparently random sequence:

CDMA reception:

1. all satellites transmit on the same carrier frequency
2. each satellite has a unique (known) code sequence
3. correlating the received (noisy) signal with each code yields a coherent energy accumulation peak when the pattern is detected in the signal

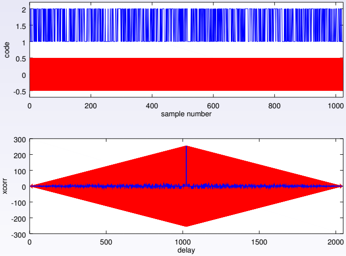


+ **magnitude** of the cross-correlation indicates whether a bit is found

Pulse compression basics

- The longer the code (T), the longer the time during which the integral of xcorr accumulates energy and **smoothes noise**,
- but long pulse induces **loss of time resolution** $\Rightarrow$ cross-correlation is a broad peak
- strong variation of code over time $\Rightarrow$ increased bandwidth $B \Rightarrow$ cross correlation peak width $1/B$

pulse compression ratio (PCR) $= B \cdot T$



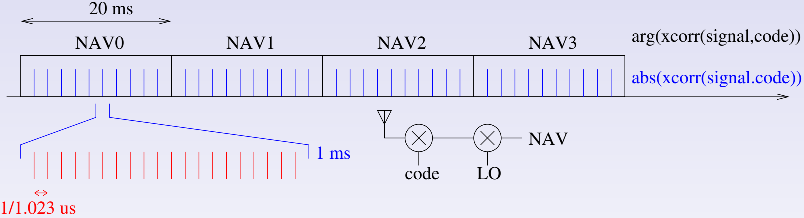
Remember: GPS is designed for **timing signals** with better than one "chip" resolution.

```
noise=rand(1023,1)'*7;
noise=noise-mean(noise);
b=[1:1023];
b=mod(b,2);b=b-mean(b);
plot(xcorr(b+noise,b),'r');hold on

a=cacode(1,1);a=a-mean(a);
plot(xcorr(a+noise,a));
plot(a+1.5);hold on;plot(b,'r');
```

CDMA: decoding GPS

Modulation steps:

- the carrier is binary-phase shift keying modulated with the satellite identifier at a rate of 1.023 MHz (phase rotations 0-180°)
- the message is additionnally binary-phase shift keying modulated over the previous signal (50 bps)



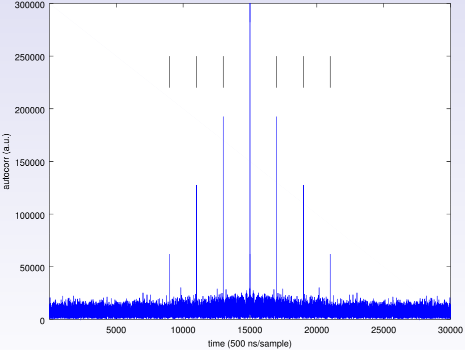
- when demodulating; first eliminate the code, ...
- ... to identify and eliminate the carrier,
- in order to recover the message.

The carrier frequency is not accurately known (Doppler shift): **what LO offset is acceptable for demodulating the message ?**

CDMA: decoding GPS

Even if we did not know the GPS encoding scheme, knowing that this code repeats is enough to assess whether a GPS signal is usable: **autocorrelation**

```
f=fopen('file.bin');d=fread(f,inf,'uchar');fclose(f);
d=d(1:2:end)-127+i*(d(2:2:end)-127);
time=[-10000:10000];
dx=abs(xcorr(d-mean(d),d-mean(d)));
plot(time,dx(2e6-10000:2e6+10000)); ylim([0 1e6]) % 2 MHz
```



$xcorr(x,y) = iFT(FT(x) \cdot FT^*(y))$

if $x = A\exp(j(\delta\omega t + \varphi))$

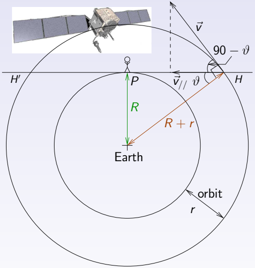
then $autocorr(x) = xcorr(x,x)$

is $iFT(|FT(A)|^2) \forall \delta\omega$

Repetition every 1 ms at 2 MS/s $\Rightarrow$ max(autocorr) every 2000 samples

CDMA: decoding GPS

- Decoding GPS is *only* possible if the carrier frequency is accurately known ...
- ... whcih can only be identified after removing the code from the received signal !
- Initial **exaustive** (*Acquisition*) search of all possible codes and frequency offsets (brute force) for later only *tracking* satellites known to be visible.
- What frequency offset should we look for ?



Doppler shift: $(R+r) = 20000 + 6400$ km in 12 h ($T^2/R^3 = \text{cst}$) $\Rightarrow |\vec{v}| = 3830$ m/s

Since $\sin(\theta) = \frac{R}{r+R}$ or $R \simeq 6400$ km

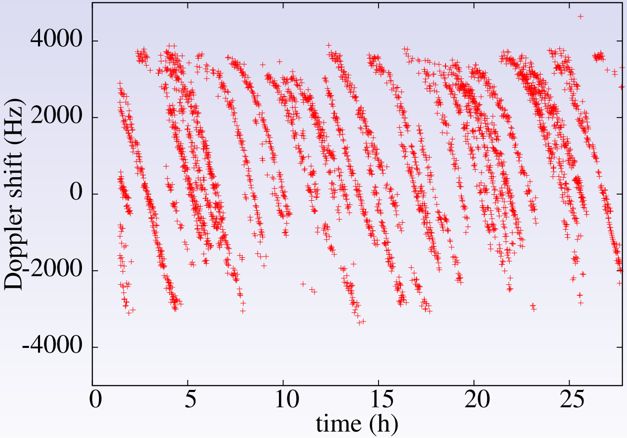
$\Rightarrow |\vec{v}_{//}| = |\vec{v}|\cos(90-\theta) = |\vec{v}|\sin(\theta) = |\vec{v}|\frac{R}{r+R}$

Result: $|\vec{v}_{//}| \in [\pm 4880]$ Hz
+ local oscillator contribution (bias and random fluctuations) !

Application: decode an acquired signal, using the GPS pseudo-random code generator available at `fr.mathworks.com/matlabcentral/fileexchange/14670-gps-c-a-code-generator/`

Observed Doppler shift

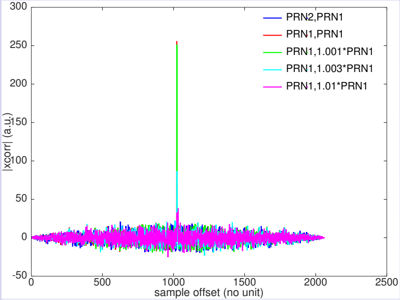
Record Doppler offset provided by `gnss-sdr` as a function of time for all visible satellites

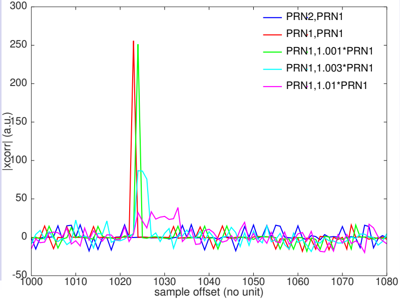


Doppler indeed $\in [\pm 4000]$ Hz accounting for minimum elevation for detectable signal

Doppler analysis frequency step

How accurately should the Doppler shift be known ?

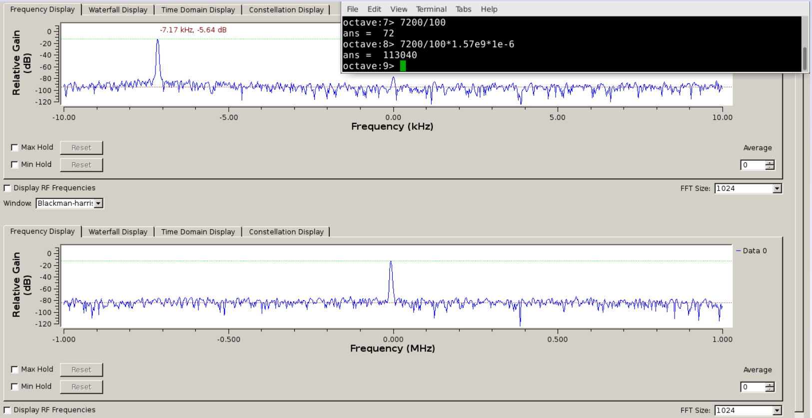




- 1023 kb/s $\simeq$ 1 μs/bit
- 1 ms long sentence: if the last bit mismatches: $dt/t = 10^{-6}/10^{-3} = 10^{-3}$
- $df/f = dt/t \Rightarrow df =$ $10^{-3} \times 1023$ kb=1 kHz
- to be safe, we select df=500 Hz

On the need for high stability LO: offset v.s Doppler

Recording a 100 MHz carrier referenced to a Cs clock:

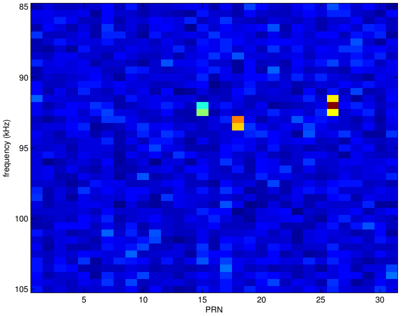


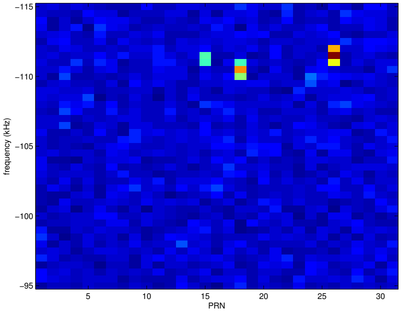
-75 ppm offset or 120 kHz at 1.57 GHz

$\Rightarrow$ rather than **20** Doppler frequencies ($\pm$5 kHz with 500 Hz steps) we must probe $\geq$ **500** Doppler frequencies

CDMA: decoding GPS

Why do we need accuracte oscillators ?





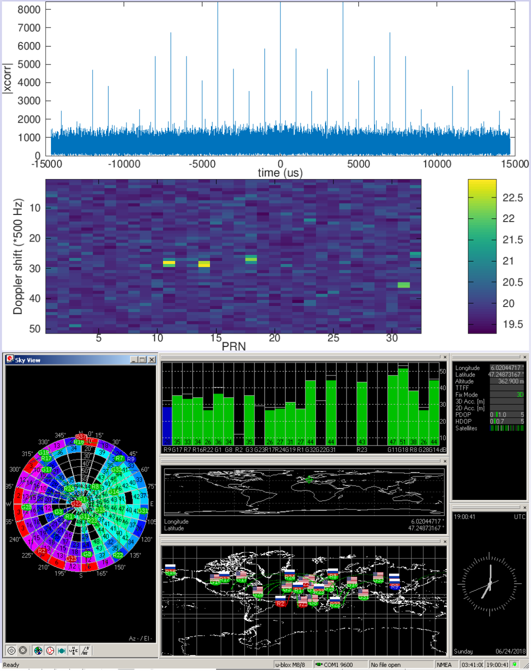
E4000 DVB-T
+59 ppm bias
=+91 kHz at 1575.42 GHz

R820T DVB-T
-68 ppm bias
=-107 kHz at 1575.42 GHz

Instead of searching a ± 5 kHz range (Doppler) with 500 Hz steps, we must search ± 150 kHz range $\Rightarrow$ computation time[8] multiplied by 30 !

[8] 20 kHz range with 500 Hz steps on $2 \cdot 10^5$ samples: 302 seconds with Matlab R2010, 342 seconds with GNU/Octave 3.8.2

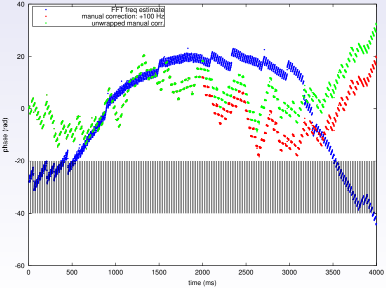
SDR v.s U-Blox

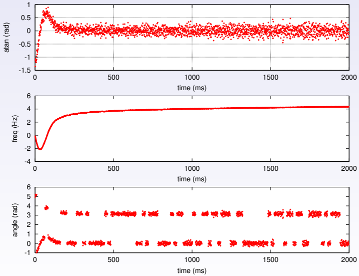


SV 10, 20, 27, 32 best visible with both receivers recording at the same time

CDMA: decoding GPS

- Cross-correlating the received RF signal with orthogonal codes allows for identifying the source of the signal, but the message is lost
- once the **acquisition** phase is completed, **tracking** by controlling LO on the received carrier
- challenge: the phase is used both to encode the message and track the carrier
- how to eliminate the phase modulation to control the frequency ?
- N-PSK : $\varphi^N = 0[2\pi]$ but reduction by a factor N of the allowed frequency offset



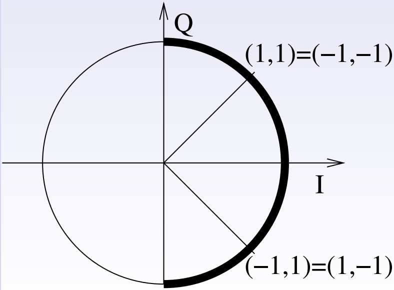


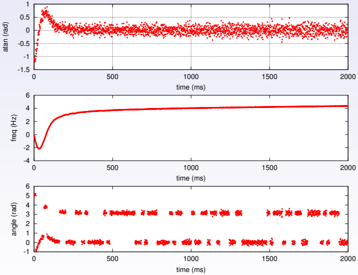
Time transfer and Global Navigation Satellite Systems (GNSS) spoofing, & spoofing detection

J.-M Friedt & al.

CDMA: decoding GPS

- Cross-correlating the received RF signal with orthogonal codes allows for identifying the source of the signal, but the message is lost
- once the **acquisition** phase is completed, **tracking** by controlling LO on the received carrier
- challenge: the phase is used both to encode the message and track the carrier
- how to eliminate the phase modulation to control the frequency ?
- atan(Q/I) v.s atan2(Q,I): Q/I cannot detect 180° phase rotation, while atan2 provides NAV..



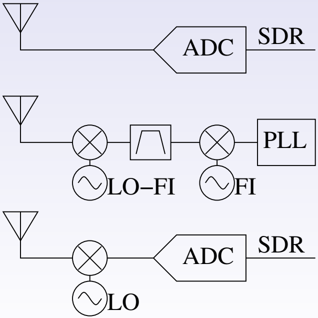


GNSS basics: jamming & spoofing

- Low received power ⇒ emitting on 1575.42 MHz will jam the signal, trivial to detect (loss of service), no technical challenge
- Spoofing: creating unwanted signal ⇒ user believes the service is still active, but erroneous information
- Spoofing used to be restricted to high grade, expensive equipment → software defined radio: $<$ 200 euro experiment

Analyzing RINEX is **too late** (processed data): software defined radio GNSS receiver to access the raw radiofrequency information (I/Q stream) ⇒ focus on the received radiofrequency signals

Software defined radio: early analog to digital conversion for digital processing of the radiofrequency signals



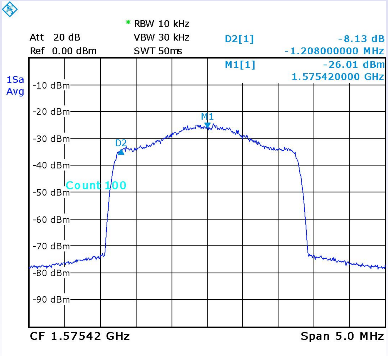
SDR spoofing demonstration

- Analog Devices PlutoSDR: AD9363 (70–6000 MHz) frontend + Zynq SOC
- Collect NAV ephemeris from the internet [9] to simulate existing constellation (period = 12 h ⇒ NAV will be valid for a couple of hours)
- Generate NAV messages for the satellites in view of the receiver (spoof for a region not too far from real location)
- Emit the signal at a level reasonably close but stronger than real signal
- Works easily with mobile phone/consumer electronics
- Insufficient LO stability for higher grade GPS (e.g. cars): replace TCXO with proper OCXO for long term stability

[9] constellation characteristics ⇒ location independent

Spoofing tools

- PlutoSDR emitter : 0 dBm output spread over 2 MHz bandwidth (1023 Mb/s) $\Rightarrow$ 30 dB peak power drop
- Software [10] running on the host PC synthesizing the I/Q coefficients streamed to the modulator, generating navigation messages representative of the simulated constellation (Zynq does not seem powerful enough for real time I/Q generation)



Range of the attack:
RX power [1]
$P_{rcv} \geq -130 + 6$ dBm
TX power=-30 dBm
FSPL @ 1575.42 MHz
$= 20\log_{10}(d) + 36$ dB
$\Rightarrow$ -124=-30-FSPL
$\Leftrightarrow 94 = 20\log_{10}(d) + 36$
$d \leq 10^{(94-36)/20} = 800$
$\Rightarrow d \leq 800$ m @ 0 dB
$\Rightarrow d \leq 80$ m @ -20 dB

[1] Global Positioning System Standard Positioning Service Signal Specificiation, p.14 (1995)

[10] github.com/Mictronics/pluto-gps-sim based on Takuji Ebinuma's github.com/osqzss/gps-sdr-sim

Mobile phone spoofing demonstration

- Find current GPS date (`sopac.ucsd.edu/convertDate.shtml`)
- Fetch satellite characteristics (RINEX navigation messages) from IGS (hourly update `hourDDD0.YYn.Z` at `ftp://cddis.gsfc.nasa.gov/gnss/data/hourly/YYYY/DDD/`)
- spoof not too far from current location to match constellation `pluto-gps-sim -e hour2110.18n -A -20.0 -t 2018/07/30,10:00:00 -l 48.3621221,-4.8223307,100`

Mostly works, but sometimes not ...

U-Blox receivers: some timid protection attempt

Unrealistic Doppler shift [11] or receiver power detection:

UBX - RXM (Receiver Manager) - RAWX (Multi-GNSS Raw Measurement Data)

Local Time 2010:144064.999000000 [s]

Leap seconds 18 (VALID) [s] Clock reset

SV	Sig...	...	Pseudo Range [...	Carrier Phase [c...	Doppl...	Lock ...	SNR	PR St...	CP St...	DO St...	P...	C...	...
G01	L1C/A	-	21042512.29	110579273.47	2331.2	28987	49	0.32	0.004	0.128	Y	Y	Y
G03	L1C/A	-	23431400.05	123132955.18	3769.9	28987	44	0.32	0.004	0.128	Y	Y	Y
G08	L1C/A	-	20490182.53	107676768.65	-1288.6	28987	51	0.32	0.004	0.128	Y	Y	Y
G10	L1C/A	-	22806998.99	119851706.37	-2822.2	27987	46	0.32	0.004	0.128	Y	Y	Y
G11	L1C/A	-	20335279.95	106862748.68	2071.6	28987	51	0.32	0.004	0.128	Y	Y	Y
G14	L1C/A	-	22487088.46	118170573.16	2378.6	29549	47	0.32	0.004	0.128	Y	Y	Y
G18	L1C/A	-	19723350.96	103647037.50	1000.7	28987	52	0.32	0.004	0.128	Y	Y	Y
G20	L1C/A	-	25254720.41	132714563.55	-3309.7	30549	42	0.64	0.004	0.256	Y	Y	Y
G22	L1C/A	-	21696336.75	114015144.79	2757.1	28987	48	0.32	0.004	0.128	Y	Y	Y
G27	L1C/A	-	22445151.02	117950185.18	-3083.7	27987	47	0.32	0.004	0.128	Y	Y	Y
G28	L1C/A	-	23200644.74	121920339.76	1196.3	29549	46	0.32	0.004	0.128	Y	Y	Y
G32	L1C/A	-	22104258.90	116158785.54	871.1	27987	48	0.32	0.004	0.128	Y	Y	Y

"Accurate" (hydrogen maser controlled) synthesizer clocking the PlutoSDR with a 40 MHz source

Frequency shifted 40 MHz-200 Hz source (5 ppm): spoofing is detected but the U-Blox still keeps on streaming position information

UBX - RXM (Receiver Manager) - RAWX (Multi-GNSS Raw Measurement Data)

Local Time 2010:144025.001000000 [s]

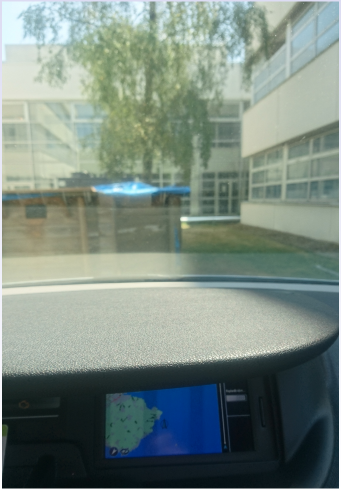
Leap seconds 18 (VALID) [s] Clock reset

SV	Sig...	...	Pseudo Range [...	Carrier Phase [c...	Doppl...	Lock ...	SNR	PR St...	CP St...	DO St...	P...	C...	...
G01	L1C/A	-	21595489.84	113485089.53	-5534.1	5159	49	0.32	0.004	0.128	Y	Y	N
G08	L1C/A	-	21015648.45	110437999.70	-9143.4	5159	51	0.32	0.004	0.128	Y	Y	N
G10	L1C/A	-	23320737.35	122551318.75	-10690.8	5159	45	0.32	0.004	0.128	Y	Y	N
G11	L1C/A	-	20886305.62	109758300.25	-5787.5	5159	51	0.32	0.004	0.128	Y	Y	N
G14	L1C/A	-	23040448.59	121078390.43	-5480.9	5159	47	0.32	0.004	0.128	Y	Y	N
G18	L1C/A	-	20266226.13	106499756.53	-6858.3	5159	51	0.32	0.004	0.128	Y	Y	N
G20	L1C/A	-	25764711.64	135394486.83	-11187.4	5159	42	0.32	0.004	0.128	Y	Y	N
G22	L1C/A	-	22252567.48	116938055.93	-5105.4	5159	48	0.32	0.004	0.128	Y	Y	Y
G27	L1C/A	-	22956908.07	120639385.55	-10949.7	5159	47	0.32	0.004	0.128	Y	Y	N
G28	L1C/A	-	23745024.53	124780962.75	-6658.2	5159	45	0.32	0.004	0.128	Y	Y	N
G32	L1C/A	-	22646175.70	119006481.65	-6980.4	5159	47	0.32	0.004	0.128	Y	Y	N
G03	L1C/A	-	23995311.71	126096225.25	-4099.7	5159	45	0.32	0.004	0.128	Y	Y	N

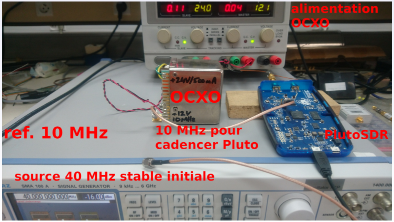
[11] orbit @ 20000 km above the Earth surface in 12 h $\Rightarrow$ 3840 m/s tangential velocity $\Rightarrow$ maximum $v = 3840 \times 6400/(6400 + 20000) = 930$ m/s towards the receiver or a Doppler shift $f_0 \times v/c \leq 4.9$ kHz @ $f_0 = 1575.42$ MHz

Beyond mobile phone: cars

- Compensating for Doppler shift by providing an "ideal" reference source allows for spoofing cars, even outdoor
- Need to match the existing constellation: not too far, not too long ago (here with hydrogen maser controlled 40 MHz synthesizer)

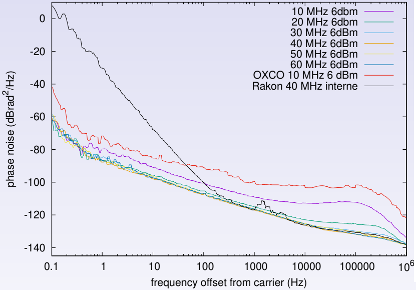


Tested on Renault & Mercedes cars

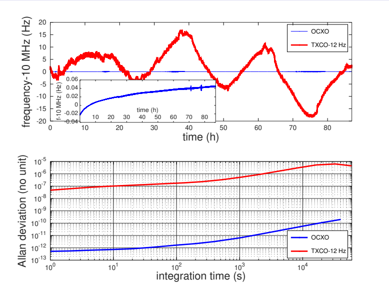


Embedded solution: replacement of the 40 MHz TCXO with a 10 MHz OCXO

Oscillator stability: short term v.s long term stability (phase noise v.s Allan deviation)



Phase noise with carrier frequency

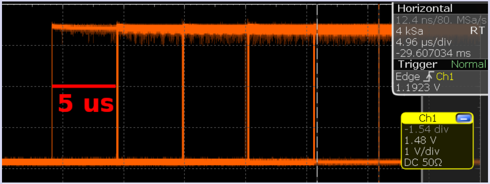


TCXO v.s OCXO

Much improved long term stability but degraded short-term stability (>100 Hz from carrier) ⇒ ideally, generate a clean 40 MHz from the 10 MHz reference

Beyond cars: timing signal

Many high-grade oscillators rely on GPS for long-term stabilization ("radio-controlled watches")



Never actively tune an atomic clock: measure offset and drift and share information with user
⇒ time offset defined by a constant (AF0), linear (AF1) and quadratic (AF2) offset.
⇒ **dynamically change these parameters** in the NAV messages of all satellites

```
  clk[0] = eph.af0 + tk * (eph.af1 + tk * eph.af2) + relativistic - eph.tgd;
  clk[1] = eph.af1 + 2.0 * tk * eph.af2;
...
  // Subframe 1
...
  sbf[0][5] = 0UL;
  sbf[0][6] = (tgd & 0xFFUL) << 6;
  sbf[0][7] = ((iodc & 0xFFUL) << 22) | ((toc & 0xFFFFUL) << 6);
  sbf[0][8] = ((af2 & 0xFFUL) << 22) | ((af1 & 0xFFFFUL) << 6);
  sbf[0][9] = (af0 & 0x3FFFFFUL) << 8;
```

is updated with

```
for (i = 0; i < MAX_CHAN; i++) { // Generate new subframes if allocated
    if (chan[i].prn != 0)
      {eph[ieph][chan[i].prn -1].af0 += 5 * pow(10,-6);    // add 5 us to AF0 every 2 mins
       eph2sbf(eph[ieph][chan[i].prn - 1], ionoutc, chan[i].sbf);
      }
}
```

Spoofing detection [12]

- Detect excessive power or unrealistic Doppler shifts: U-Blox receivers
- Proper signal generation will fool such strategies
- Our approach: analyze the raw RF signal for unrealistic characteristics

A constellation is spatially distributed, a spoofer is located at a single point ⇒ antenna array measurement and angle of arrival measurement

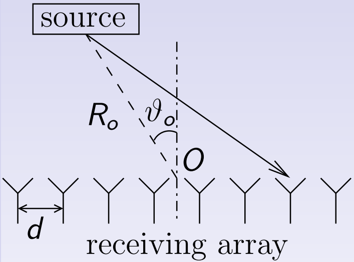
[12] R.G. Hartman, *Spoofing detection system for a satellite positioning system*, Patent US5557284A (1995):
A **pair of antennae** *in combination with a GPS signal receiver system is employed for detecting the reception of satellite information signals from a spoofing signal transmitter as opposed to those satellite information signals transmitted aboard each of the satellite vehicles which form a satellite positioning system. As described herein, an indication of the* **pointing angle between the antennae and the actual transmitter** *transmitting the satellite information signals is detected. The pointing angle and/or alternatively the range difference may be observed by monitoring the behavior of the pseudo random code associated with the carrier of the satellite information signal, or the carrier itself. In turn, pseudo range measurements, ...*

Codeless analysis

- Code analysis is computationally intensive but provides complete receiver architecture
- Spoofing detection: only analyze $s^2(t)$ which **removes** the BPSK message
- Satellite identification by different Doppler shift
- Two antennas: add a geometrical term to phase (delay of arrival)
- Direction of arrival by phase difference between antennas (cancels the Doppler from a same satellite): $arg(FFT(s_n^2))$ at the nth antenna

Solution demonstration

Power can be tuned, but **direction of arrival** will be difficult to simulate
$\Rightarrow$ replace single receiving antenna with array for phase analysis [13]



- In the far field ($R_0 \gg 2(Kd)^2/\lambda$) and narrowband ($B \ll c/(Kd)$) approximations, a plane wave hits the antenna array
- the phase shift between elements is $2\pi kd \sin\vartheta/\lambda_0$ for the kth element
- the various satellites with diffferent elevations and azimuth exhibit different ϑ_0 and their signal contribution can be separated by analyzing the phase between antennas of the array

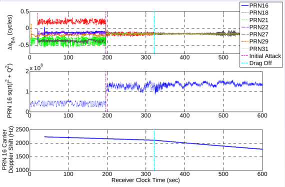


Fig. 15. Indicators of initial capture and drag-off during Libya spoofing attack, as measured by the spoofing detection receiver.

M.A. Psiaki *& al.*, *GNSS Spoofing Detection using Two-Antenna Differential Carrier Phase*, Proc. Radionavigation Laboratory Conference (2014), cited in R. T. Ioannides, T. Pany, & G. Gibbons, *Known vulnerabilities of global navigation satellite systems, status, and potential mitigation techniques*, Proc. IEEE **104** (6), 1174–1194 (2016). Top=phase, middle=power, bottom=Doppler

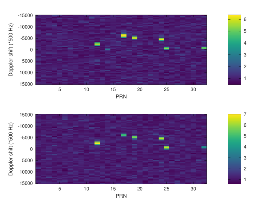
[13] $\lambda = 19$ cm $\Rightarrow K = 8$ & $d = \lambda/2 \Rightarrow Kd = 76$ cm: $R_0 > 6$ m & $B \ll 400$ MHz !

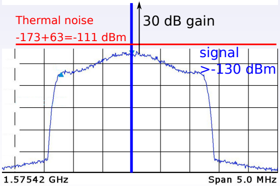
- Ettus Research B210 provides **two synchronous inputs**: bias T to GPS antenna
- Each antenna a detects the sum of all satellite n signals

$$x_{n,a} \propto \exp\left(j(\underbrace{\delta\omega_n}_{Doppler} + \underbrace{\varphi_n}_{BPSK} + \underbrace{\varphi_{n,a}}_{geometry})\right)$$

  with φ_n the PRN+NAV sequence (spectrum spreading)
- Here we do not care about PRN+NAV $\Rightarrow$ no need for the time consuming Doppler-PRN map calculation + DLL, but yet we need to **get rid of the BPSK modulation** since ...
- GPS signal is below thermal noise $\Rightarrow$ getting rid of the modulation by squaring (**low computation requirement**): $2\varphi_n = 2\{0,\pi\} = 0\ [2\pi]$ rises the signal by $10\log_{10}(1023) = 30$ dB
- Each Fourier transform peak is at $\delta\omega_n$ so the Doppler shift identifies SV n
- arg of $FFT(x_{n,1}^2)$ is $\delta\omega_n + \varphi_{n,a}$ so that $x_{n,1}^2 - x_{n,2}^2$ is $\varphi_{n,a}$ only dependent on **satellite position**
- if all $arg(FFT(x_{n,1}^2)) - arg(FFT(x_{n,2}^2))$ are equal, meaning all satellites are at the same position, we are being spoofed. A real constellation will have all phases different.

Experimental setup





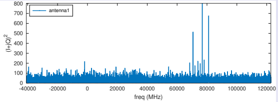
Pulse compression

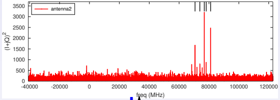
- Ettus Research B210 provides **two synchronous inputs**: bias T to GPS antenna
- Each antenna a detects the sum of all satellite n signals

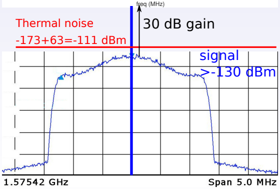
$$x_{n,a} \propto \exp\left(j(\underbrace{\delta\omega_n}_{Doppler} + \underbrace{\varphi_n}_{BPSK} + \underbrace{\varphi_{n,a}}_{geometry})\right)$$

  with φ_n the PRN+NAV sequence (spectrum spreading)
- Here we do not care about PRN+NAV $\Rightarrow$ no need for the time consuming Doppler-PRN map calculation + DLL, but yet we need to **get rid of the BPSK modulation** since ...
- GPS signal is below thermal noise $\Rightarrow$ getting rid of the modulation by squaring (**low computation requirement**): $2\varphi_n = 2\{0,\pi\} = 0\ [2\pi]$ rises the signal by $10\log_{10}(1023) = 30$ dB
- Each Fourier transform peak is at $\delta\omega_n$ so the Doppler shift identifies SV n
- arg of $FFT(x^2_{n,1})$ is $\delta\omega_n + \varphi_{n,a}$ so that $x^2_{n,1} - x^2_{n,2}$ is $\varphi_{n,a}$ only dependent on **satellite position**
- if all $arg(FFT(x^2_{n,1})) - arg(FFT(x^2_{n,2}))$ are equal, meaning all satellites are at the same position, we are being spoofed. A real constellation will have all phases different.

Experimental setup

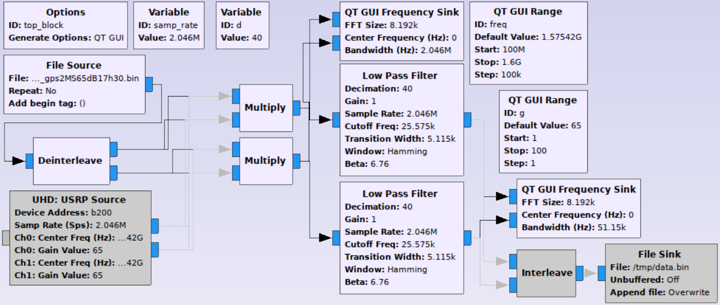






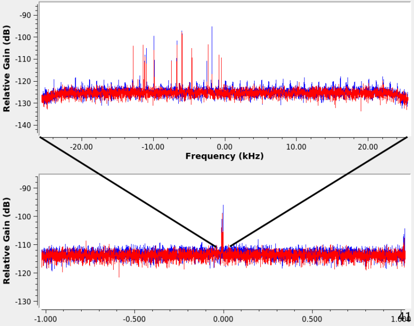
Pulse compression

Solution demonstration



GNU Radio flowgraph to reduce datarate (2-channels @ 2.046 MS/s complex float = 33 MB/s $\rightarrow$ decimate by 40 for $<$1 MB/s)

GNU/Octave: identify each peak ($\delta\omega_n$ for each satellite n) on magnitude and compute phase of each signal at this position

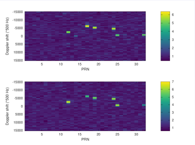


Spoofing signal datasets: clustered phase values whatever the considered satellite

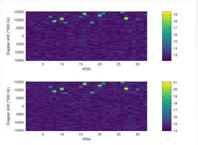
filename	phase	phase	phase	phase	phase	phase
190130_spoof2MS65dB16h03.bin	4.07	4.09	4.07	4.01	4.04	4.06
190130_spoof2MS65dB16h03.bin, segment 2	4.06	4.08	4.03	3.98	4.07	4.05
190130_spoof2MS65dB16h03.bin, segment 3	4.13	4.12	4.13	4.12	4.14	4.09
190130_spoof2MS65dB16h08.bin	3.21	3.09	3.09	3.45	3.20	
190130_spoof2MS65dB16h08.bin, segment 2	3.27	3.25	3.24	3.19	3.20	3.51
190130_spoof2MS65dB16h08.bin, segment 3	3.32	3.35	3.29	3.35	3.32	
190130_spoof2MS65dB16h08.bin, segment 4	3.45	3.53	3.51	3.46	3.46	3.53
190130_spoof2MS65dB16h15.bin	3.39	3.41	3.41	3.40	3.371	3.45
190130_spoof2MS65dB16h15.bin, segment 2	3.53	3.44	3.43	3.46	3.48	3.32
190130_spoof2MS65dB16h15.bin, segment 3	3.42	3.40	3.43	3.27	3.28	3.21
190130_spoof2MS65dB16h15.bin, segment 4	3.07	3.07	3.30	3.08	2.99	

Genuine GPS constellation datasets: phases varying in the $[-\pi, \pi]$ range

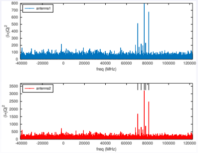
filename	phase	phase	phase	phase	phase	phase
190130_gps2MS65dB17h22.bin	3.63	0.085	-0.73	0.04	0.93	-0.53
190130_gps2MS65dB17h30.bin	-0.41	-1.14	5.74			
190130_gps2MS65dB18h41.bin	-3.22	-3.71	2.45	2.83	-4.74	
190130_gps2MS65dB18h43.bin	-2.21	1.72	0.96	-0.52		
190130_gps2MS65dB19h06.bin	2.056	-3.51	0.43	1.86	1.59	1.08
190130_gps2MS65dB19h11.bin	1.34	1.82	-1.51	0.60	2.57	
190130_gps2MS65dB19h15.bin	1.26	4.56	1.77			



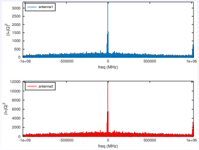
GPS constellation



Spoofed signal (Rakon)



Zoom on ...



squared signal

Solution demonstration: 2-antennas

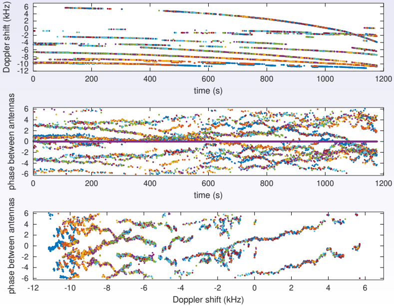
Each antenna a detects the sum of the satellite n signals

$$x_{n,a} \propto \exp\left(j(\underbrace{\delta\omega_n t}_{Doppler} + \underbrace{\varphi_n}_{BPSK} + \underbrace{\varphi_{n,a}}_{geometry})\right), \varphi_n \in [0,\pi]$$

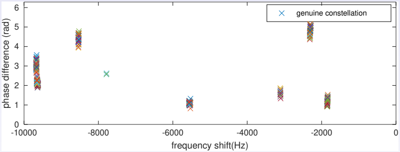
$$\Rightarrow arg(FFT(x_{n,1}^2)) - arg(FFT(x_{n,2}^2)) = \varphi_{n,2} - \varphi_{n,1}$$

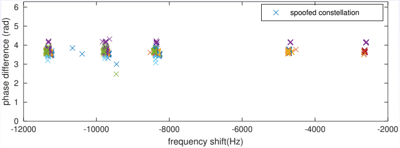
only dependent on antenna geometry and satellite position

Long term Doppler & phase monitoring



Genuine constellation

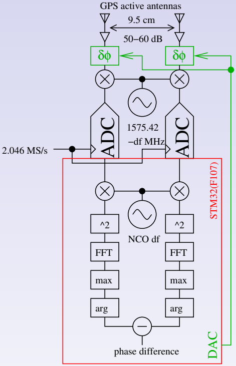




Spoofed constellation

Beamforming

- Spoofing detection is possible ⇒ mitigation [14] ?
- The ground-based signal is stronger than the spaceborne signal
  ⇒ tune antenna radiation pattern so that a null (low reception sensitivity) is directed towards spoofing source
- Also applicable to jamming mitigation
- Antenna array with the "proper" phase conditions between elements will cancel (destructive interference) the signal coming from a given direction



[14] R. Heue, *GNSS Jamming and Spoofing: Hazard or Hype?* at https://www.space-of-innovation.com/gnss-jamming-and-spoofing-hazard-or-hype/ (June 2018): "A vital means to counter jammers is the use of an **array antenna, which either is able to steer the radiation pattern of the array to form a spatial null towards the jammer**, or to provide additional gain towards the satellites. However, such controlled radiation pattern antennas (CRPA) are military technology and the availability for civil users is an exception."

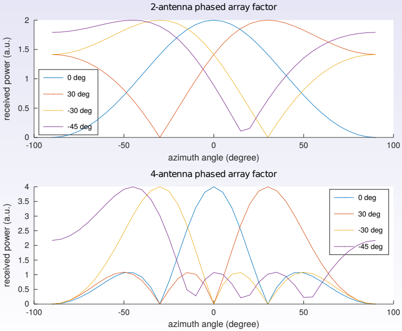
Beamforming

- Spoofing detection is possible ⇒ mitigation ?
- The ground-based signal is stronger than the spaceborne signal ⇒ tune antenna radiation pattern so that a null (low reception sensitivity) is directed towards spoofing source
- Also applicable to jamming mitigation
- Antenna array with the "proper" phase conditions between elements will cancel (destructive interference) the signal coming from a given direction

Phased array factor

$$\sum_{n=0}^{N-1} \exp(jn\Psi),\ \Psi = kd\cos(\vartheta) + \beta$$

where $k = \frac{2\pi}{\lambda}$, d distance between antennas, ϑ incidence angle and β the phase between array elements



Conclusion

- Spoofing GPS is a good opportunity to demontrate **detailed understanding** of the communication and location mechanisms
- **Ease** of implementation spoofing using (PlutoSDR) SDR implementation
- Requires a "good" stability **reference oscillator** (external source to PlutoSDR)
- B210 **SDR, computationally efficient solution** demonstration: multi-antenna receiver for anti-spoofing using Direction of Arrival (DOA) analysis with respect to constellation configuration
- ⇒ **dual receiver** appoach, one for GNSS and one for spoofing detection, or deriving signals from GNSS-SDR for spoofing detection
- Promotional video of the Rohde & Schwarz SMW200A (40 k$ for 6 GHz model) @ `https://www.rohde-schwarz.com/fr/produits/test-et-mesure/generateurs-de-signaux/video-simulateur-gnss/high-end-gnss-simulation-with-the-r-s-smw200a-episode-3_232162.html`

[13] G. Goavec-Merou, J.-M Friedt, F. Meyer, *Leurrage du GPS par radio logicielle*, MISC Special Issue (2019), translated at `jmfriedt.free.fr/misc_gps_eng.pdf`

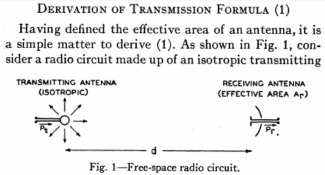
Link budget

- a radiofrequency (electrical) power is emitted, either isotropically or in a directional pattern with an antenna gain G_1: $P_E \times G_1$
- this power spreads on a sphere centered on the emitter: in the case of isotropic emitter, the area of this sphere is, at a distance d, $4\pi d^2$
- if $G_1 > 1$, then only a fraction $4\pi d^2/G_1$ of the sphere is illuminated
- this sphere intersects the receiver, which can detect any incoming signal on a 4π-steradian sphere on a typical area of λ^2
- this receiver might exhibit a reception antenna gain G_2

$$\frac{P_R}{P_E} = G_1 G_2 \left(\frac{\lambda}{4\pi d}\right)^2 : \text{Friis}^{14} \text{ equation}$$

or Free Space Propagation Loss (FSPL), since $20\log_{10}(c/4/\pi) = 147.5$ dB

$$FSPL = 20\log_{10}(f) + 20\log_{10}(d) - 147.55 \text{ dB}$$


Derivation of Transmission Formula (1)

Having defined the effective area of an antenna, it is a simple matter to derive (1). As shown in Fig. 1, consider a radio circuit made up of an isotropic transmitting

Fig. 1—Free-space radio circuit.

[14] H.T. Friis *A Note on a Simple Transmission Formula*, Proc. I.R.E. 254- (1946)

Link budget

- a radiofrequency (electrical) power is emitted, either isotropically or in a directional pattern with an antenna gain G_1: $P_E \times G_1$
- this power spreads on a sphere centered on the emitter: in the case of isotropic emitter, the area of this sphere is, at a distance d, $4\pi d^2$
- if $G_1 > 1$, then only a fraction $4\pi d^2/G_1$ of the sphere is illuminated
- this sphere intersects the receiver, which can detect any incoming signal on a 4π-steradian sphere on a typical area of λ^2
- this receiver might exhibit a reception antenna gain G_2

$$\frac{P_R}{P_E} = G_1 G_2 \left(\frac{\lambda}{4\pi d}\right)^2 : \text{Friis equation}$$

Application:

1. a GPS satellite emits 50 W (17 dBW=47 dBm) at 1575.42 MHz with an antenna gain of 13 dBi and flies at 20000 km over the Earth
2. $FSPL = 182$ dB$\Rightarrow P_R = -152$ dBW$= -$**122** dBm
3. receiver sensitivity: typically around -159 dBm (`usglobalsat.com/store/download/53/et312_ug.pdf`)
4. DVB-T: detection limit around -95 dBm (10 dB SNR) + 27 dB antenna gain = **-122 dBm** detection limit

Link budget

- a radiofrequency (electrical) power is emitted, either isotropically or in a directional pattern with an antenna gain G_1: $P_E \times G_1$
- this power spreads on a sphere centered on the emitter: in the case of isotropic emitter, the area of this sphere is, at a distance d, $4\pi d^2$
- if $G_1 > 1$, then only a fraction $4\pi d^2/G_1$ of the sphere is illuminated
- this sphere intersects the receiver, which can detect any incoming signal on a 4π-steradian sphere on a typical area of λ^2
- this receiver might exhibit a reception antenna gain G_2

$$\frac{P_R}{P_E} = G_1 G_2 \left(\frac{\lambda}{4\pi d}\right)^2 : \text{Friis equation}$$

What is the thermal noise power ?

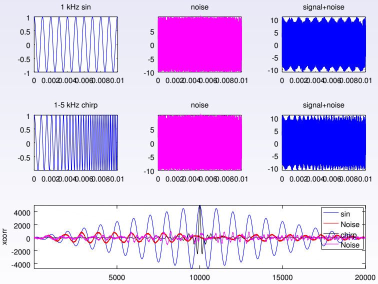
1. 1 MHz bandwidth (1023 kHz) so that $10\log_{10}(10^6) = 60$ dB
2. $-174 + 60 = -\mathbf{114}$ dBm> -122 dBm ! [14]
3. but 30 dB=1023 kHz/1 kHz pulse compression: $-122 + 30 = -\mathbf{92} > 114$ dBm ($SNR \simeq 22$ dB after compression)
4. the cross-correlation brings the signal out of the noise: a spectral analysis (FFT) **cannot display** the GPS signal !

[14] $P = 10\log_{10}(k_B T)$ with $k_B = 1.38 \cdot 10^{-23}$ J.K^{-1} & $T = 293$ K, +30 dB for mW

Pulse compression basics

- The longer the code (T), the longer the time during which the integral of xcorr accumulates energy and **smoothes noise**,
- but long pulse induces **loss of time resolution** $\Rightarrow$ cross-correlation is a broad peak
- strong variation of code over time $\Rightarrow$ increased bandwidth B $\Rightarrow$ cross correlation peak width $1/B$

pulse compression ratio (PCR) $= B \cdot T$

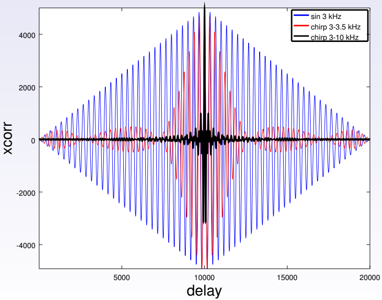



```
time=[0:1e-6:1e-2]; %samp. rate=1 us

x=chirp(time,1e3,time(end),1e3);
noise=20*rand(length(x),1)';
noise=noise-mean(noise);
xx=xcorr(x,x); xb=xcorr(x,noise);
plot(xx,'b-');hold on;plot(xb,'r-');

x=chirp(time,1e3,time(end),5e3);
xx=xcorr(x,x); xb=xcorr(x,noise);
plot(xx,'k-');hold on;plot(xb,'m-');
```

Pulse compression basics

- The longer the code (T), the longer the time during which the integral of xcorr accumulates energy and **smoothes noise**,
- but long pulse induces **loss of time resolution** $\Rightarrow$ cross-correlation is a broad peak
- strong variation of code over time $\Rightarrow$ increased bandwidth B $\Rightarrow$ cross correlation peak width $1/B$

pulse compression ratio (PCR) $= B \cdot T$




```
time=[0:1e-6:1e-2]; %samp. rate=1 us

x=chirp(time,1e3,time(end),1e3);
noise=20*rand(length(x),1)';
noise=noise-mean(noise);
xx=xcorr(x,x); xb=xcorr(x,noise);
plot(xx,'b-');hold on;plot(xb,'r-');

x=chirp(time,1e3,time(end),5e3);
xx=xcorr(x,x); xb=xcorr(x,noise);
plot(xx,'k-');hold on;plot(xb,'m-');
```