LoRa

Martin Heusse
LIG / Drakkar

Lora in the ISM bands
channel spacing : 200kHz

- 433MHz Band
  - ✓ Max Tx power 10dBm
- EU 863-870MHz Band
  - ✓ Max Tx power : 20dBm, by default 14dBm
  - ✓ Rx channels for the *gateways*

Modulation	Bandwidth [kHz]	Channel Frequency [MHz]	FSK Bitrate or LoRa DR / Bitrate	Nb Channels	Duty cycle
LoRa	125	868.10 868.30 868.50	DR0 to DR5 / 0.3-5 kbps	3	<1%

  - ✓ *Duty cycle* is computed **per sub band**
  - ✓ [LoRaWAN] All gateways listen to (up too) 8 channels in parallel (Sx1302).

Specified to the *devices* when they associate

Lora in the ISM bands
channel spacing : 200kHz (cont.)

DataRate	Configuration	Indicative physical bit rate [bit/s]
0	LoRa: SF12 / 125 kHz	250
1	LoRa: SF11 / 125 kHz	440
2	LoRa: SF10 / 125 kHz	980
3	LoRa: SF9 / 125 kHz	1760
4	LoRa: SF8 / 125 kHz	3125
5	LoRa: SF7 / 125 kHz	5470
6	LoRa: SF7 / 250 kHz	11000
7	FSK: 50 kbps	50000
8..15	RFU	

TXPower	Configuration
0	20 dBm (if supported)
1	14 dBm
2	11 dBm
3	8 dBm
4	5 dBm
5	2 dBm
6..15	RFU

- **Real world range : a few km NLOS, $\approx$ 20 km with LOS**
- Sensitivity = -140dBm (at SF12), $\Rightarrow$ link budget : $\approx$ 150 dB (same as for e.g. GSM)
- Payload max size : from 51 (@SF12) to 242 B (from *datarate* 4 and higher)

ISM 868MHz band

https://www.arcep.fr/uploads/tx_gsavis/19-0300.pdf

https://www.anfr.fr/fileadmin/mediatheque/documents/tnrbf/TNRBF_2020-03-16.pdf

Freq.	*Duty cycle*	other uses
863-865 MHz	0,1 %	Cordless microphones
865-868 MHz	1%	RFID – ??
868-868,6 MHz	1%	(802.15.4 Sub-GHz)
868.6-868,7 MHz	1%	Alarms
868,7-869,2 MHz	0.1%	
869,2-869,4 MHz	0.1%	Alarms
869,4-869,65 MHz	10% (GWs), 2.5%	500 mW! Sensing…
869,65-869,7 MHz	10%	
869,7-870 MHz	1%	

ERC Recommendation 70-03

`https://docdb.cept.org/download/25c41779-cd6e/Rec7003e.pdf`

Sub band	Freq. (MHz)	Power	*Duty cycle*	BW (MHz)
h1.4	865-868	14 dBm	1%	3
h1.5	**868-868.6**	14 dBm	**1%**	0.6
h1.6	868.7-869.2	14 dBm	0.1%	0.5
h1.7	**869.4-869.65**	27 dBm	**10%**	0.25
h1.8	869.7-870	7 dBm	100%	0.3
h1.9	869.7-870	14 dBm	1 %	0.3

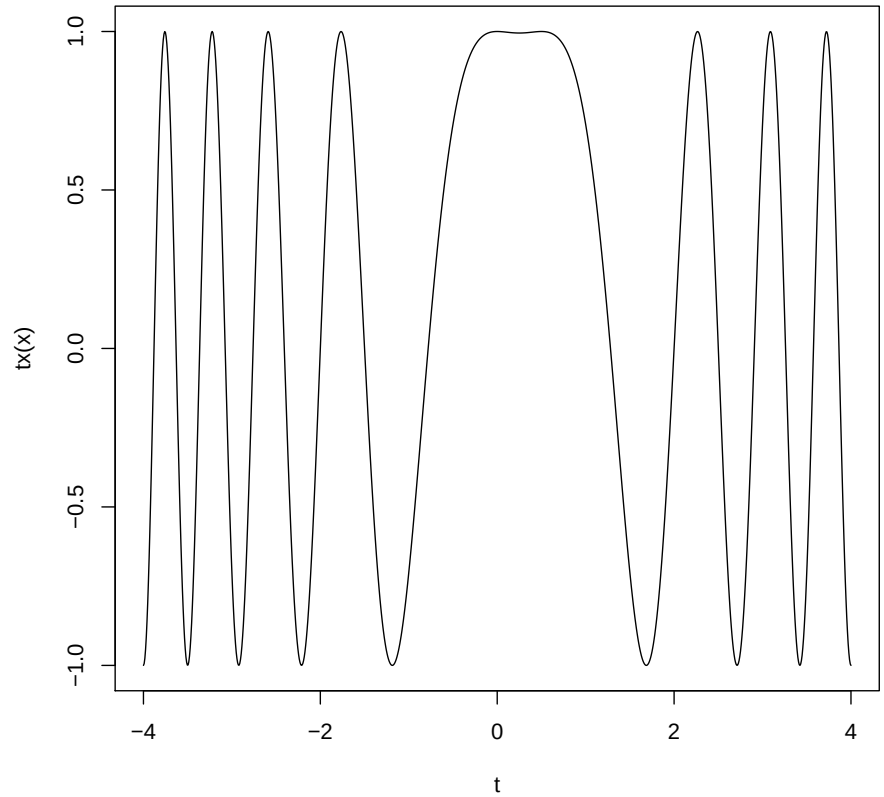
Duty cycles are computed per sub-band : a *device* may consume 1% in h1.5, 10% in h1.7, 1% in h1.4, during the same hour for instance

h1.5 encompasses the **3 defaults LoRaWAN channels**,

h1.7 is used by the GW to respond to the devices (*cf.* RX2)

What is a *chirp* ?
CSS : Chirp Spread Spectrum

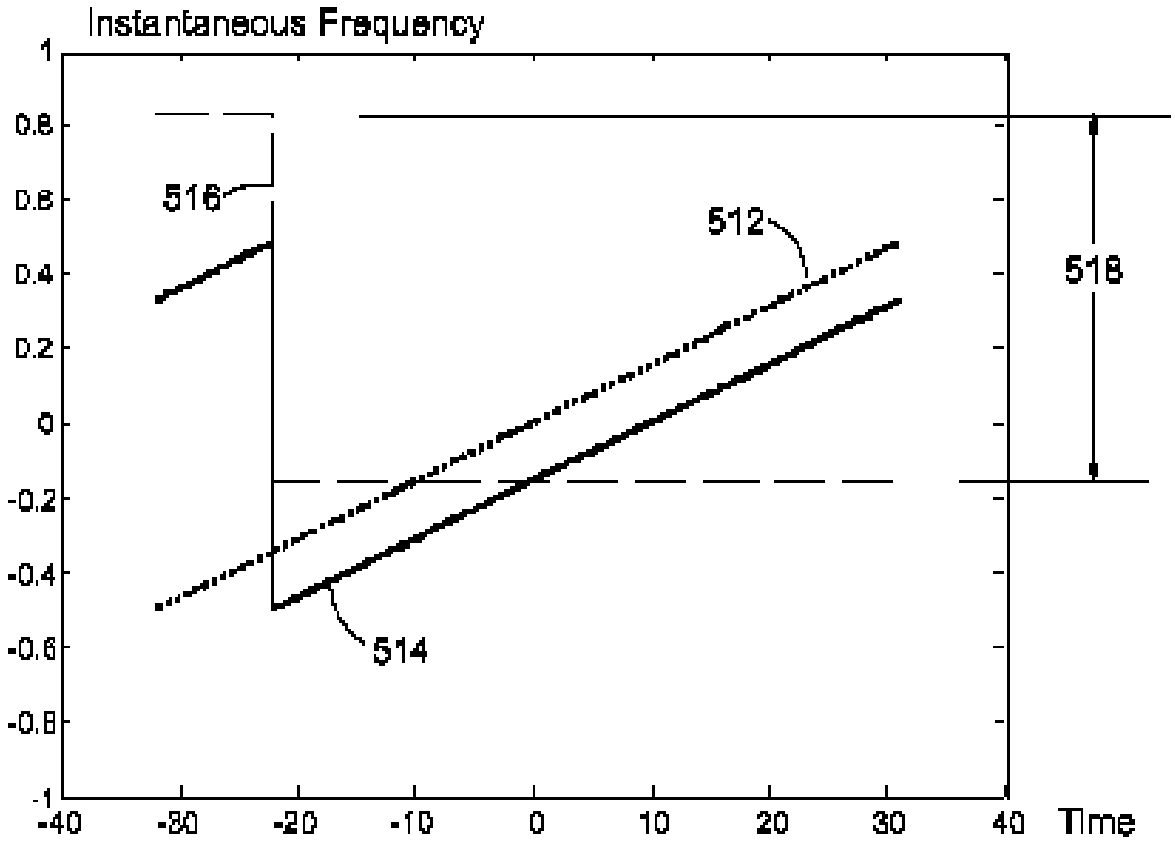
- A linear frequency sweep/ramp $-\frac{BW}{2} < f < \frac{BW}{2}$



- Used by radars, bats, dolphins…

Coding information on a chirp

- It is the start freq. offset that codes the information (line labeled 514 below)



Examples

- Example with upchirps: $SF = 2$, 4 symbols: $0, 1, 2, 3$

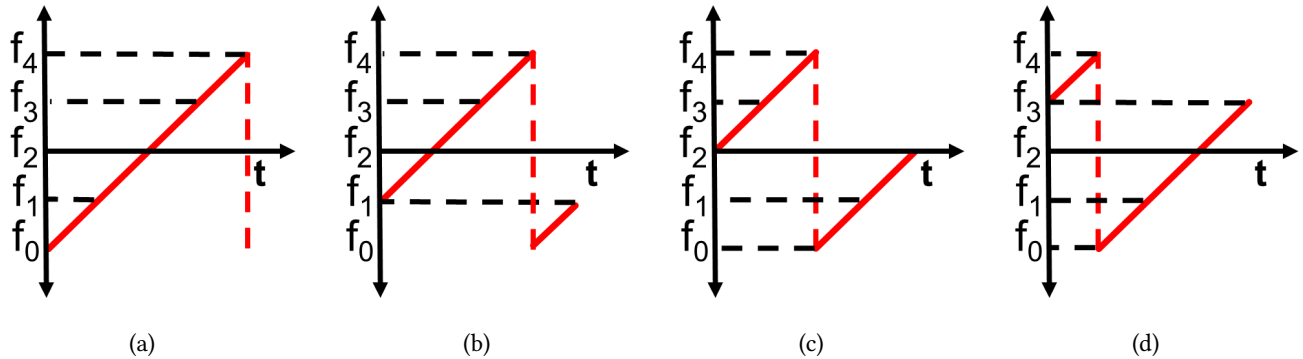
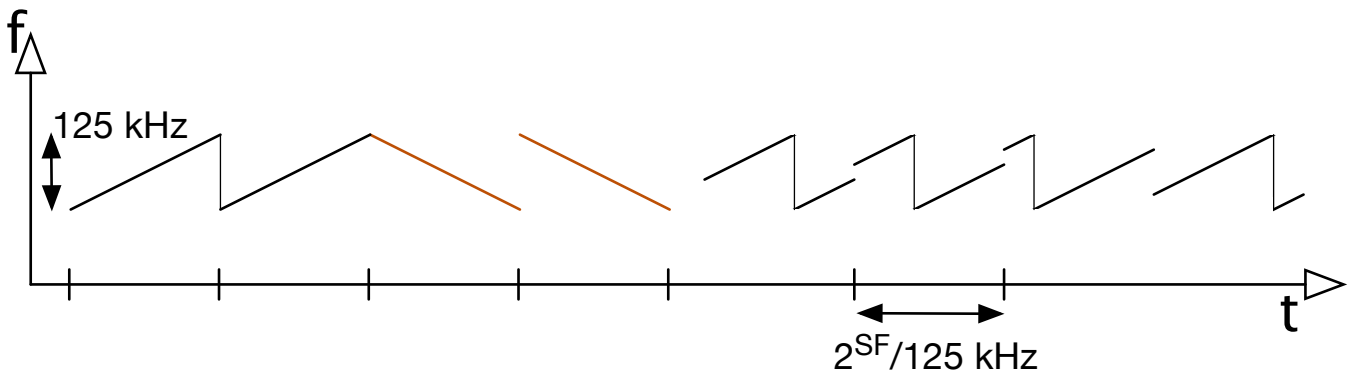


Fig. 6. Four CSS symbols when spreading factor is 2.

(LoRa Backscatter: Enabling The Vision of Ubiquitous Connectivity, V. Talla et al.)



Reception

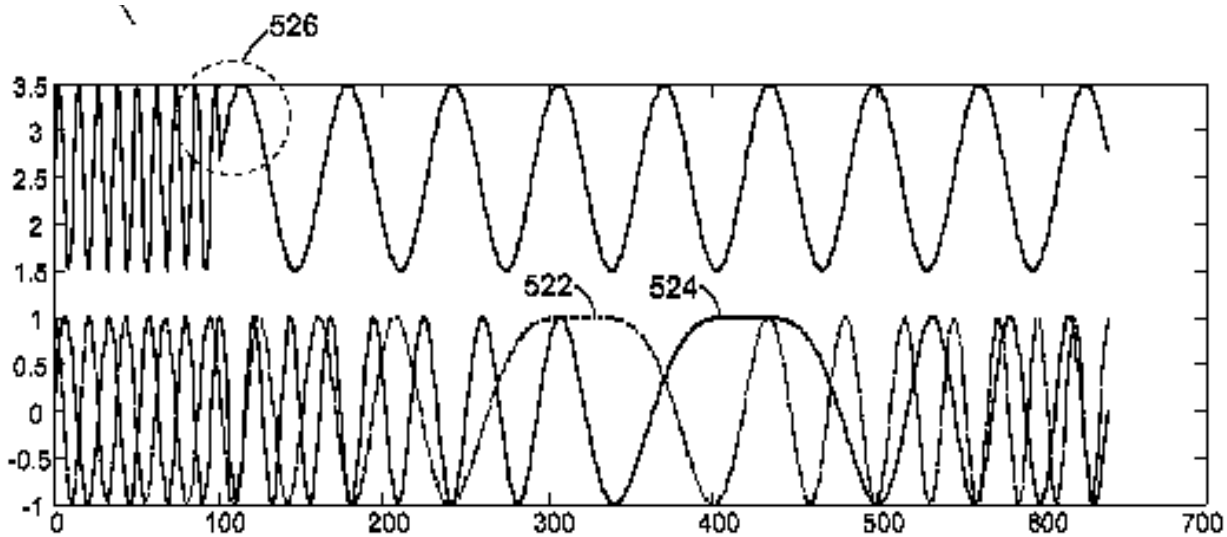
- Multiplication of rx signal with a complex conjugate chirp (down chirp)

$$e^{2\pi jt[f_0+(at+b) \bmod BW]} \times e^{-2\pi jt[f_0+(at) \bmod BW]} = e^{2\pi jt[b \bmod BW]}$$

N.B. : $a = \frac{BW^2}{2^{SF}} \rightarrow$ so it takes a time $\frac{2^{SF}}{BW}$ to sweep BW

Reception (cont.)

- if both chirps are in sync, we get a constant, otherwise:



FFT-based reception

- FFT after sampling at rate BW
  The signal occupies a band BW, it would be pointless to sample faster than this…
- The symbol duration is $\frac{2^{SF}}{BW} \rightarrow 2^{SF}$ samples
- By frequency aliasing, **a single frequency appears in the FFT** !

Spread spectrum

- Spreading factors from 7 to 12 $\Leftrightarrow$ N goes from 2^7 to 2^{12},
  7 to 12 bits per symbol, 0.05 to 0.002 b/s/Hz
- The bigger the **SF** the longer the **chirp** — 33 ms @ SF12.
  For LoRa, **the preamble is also proportional to the SF**

The actual range of spread spectrum is ≈ 20 $R_b = SF \times \frac{BW}{2^{SF}}$

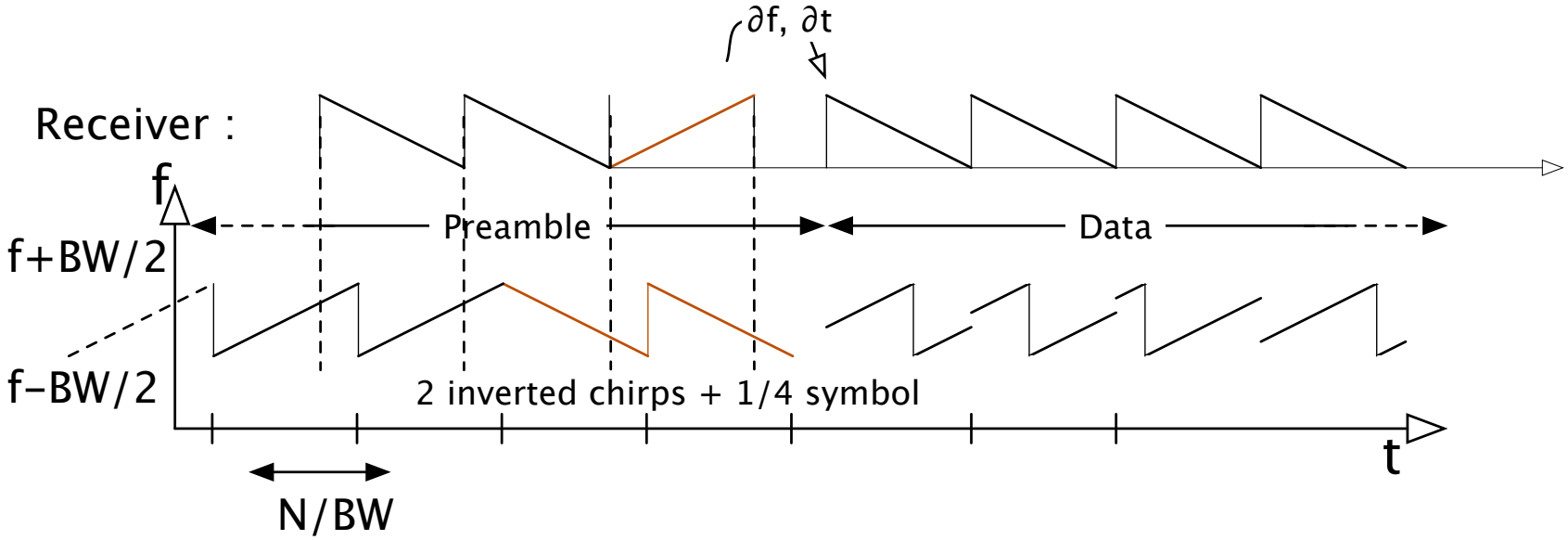
- Error correcting codes $R = \frac{4}{5}$
  at SF12, PER is still low at an SNR of -20 dB
- The actual max. SF is $\approx$ 340 ($2^{12}/12$), so a transmission may survive a collision with a node closer by a ratio of $\approx \sqrt{340}$

LoRa operating SNR

Table: LoRa physical layer parameters for 125 kHz bandwidth.

SF	DR*j*	Data rate [b/s]	PL_{max} [B]	Airtime τ_j [ms]	*SNR* limit q_j [dB]
7	DR5:	5469	230	102.7	-7.5
8	DR4:	3125	230	184.8	-10
9	DR3:	1758	123	328.7	-12.5
10	DR2:	977	59	616.5	-15
11	DR1:	537	59	1315	-17.5
12	DR0:	293	59	2466	-20

Initial Synchronisation

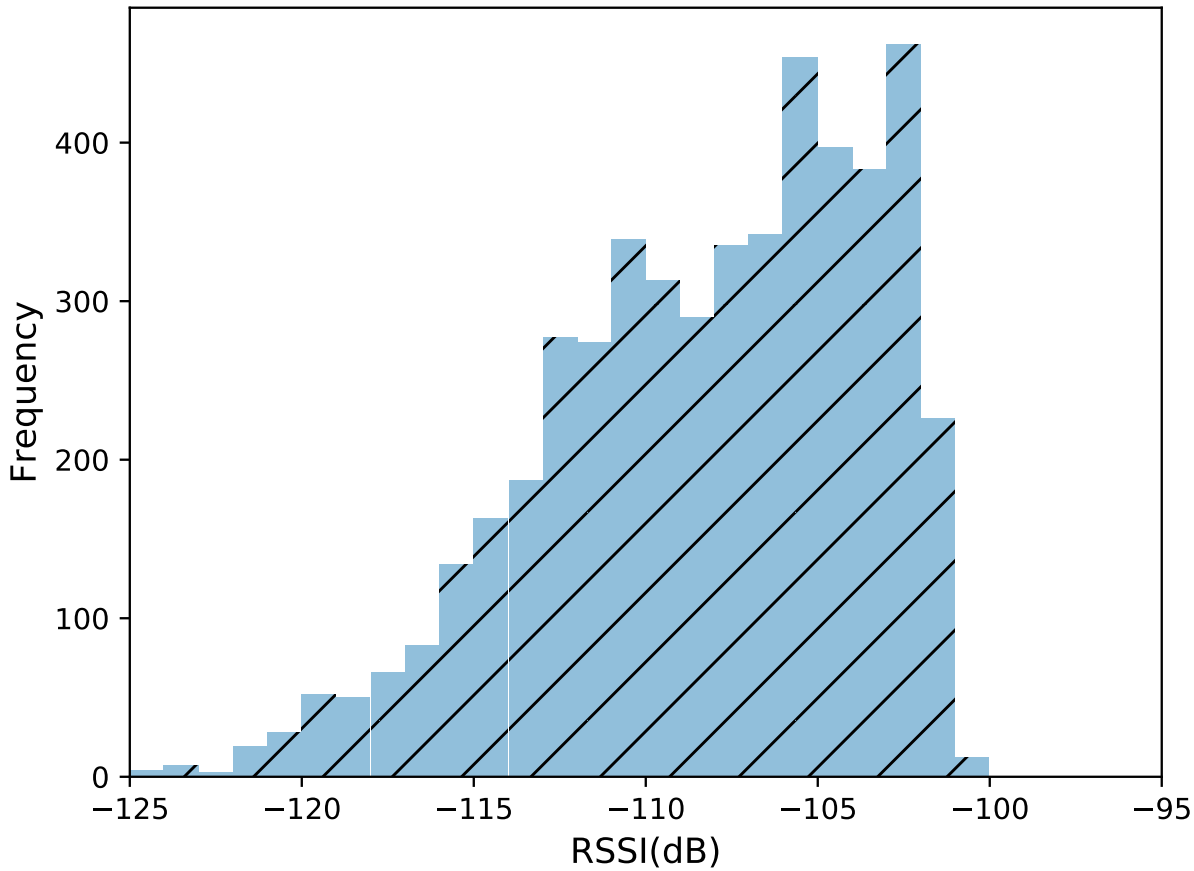


- Default preamble: 8 up-chirps + 2 up-chirps (Sync word) + 2.25 down-chirps (preamble overhead of 147 bits @SF12!; 86 bits @SF7…)(Sync word=0x34 for public networks)
- The inverted chirps in the preamble allow to find the two unknown variables: the transmitter **frequency** and the relative time **reference**
- Preamble detection takes only a couple of chirps
  → short rx1 and rx2 windows for DL class A transmissions

LoRa wireless channel

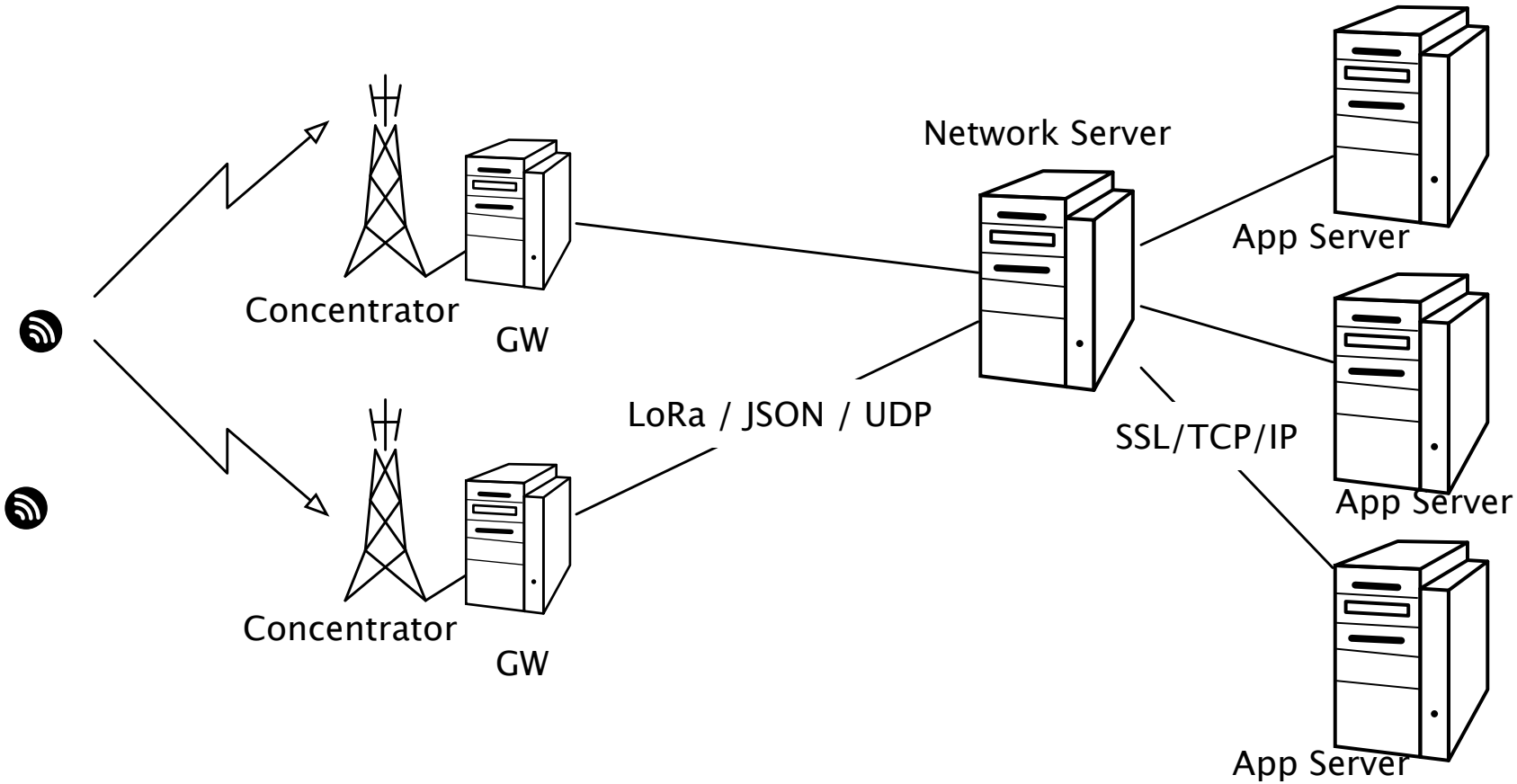
- A LoRa channel is narrower than a GSM channel (125kHz *vs.* 200+kHz)
  Channel width is the same order of magnitude as coherence band —maybe **narrower**— for a range of several km
  $\rightarrow$ changing channel changes the gain and may help, regardless of interference matters
- The chirp slope is inverted between uplink and downlink (`ipol` parameter of the GW)

LoRa wireless channel (cont.)



Typical gain distribution between a device and a GW
It varies a lot! (Rayleigh channel)(measures by T. Attia)

LoRaWAN network



LoRaWAN network (cont.)

Radio PHY layer:

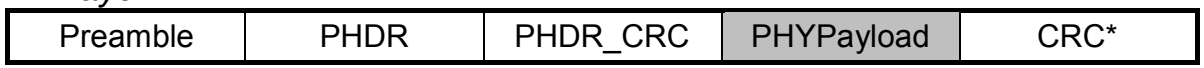


Figure 5: Radio PHY structure (CRC* is only available on uplink messages)

PHYPayload:

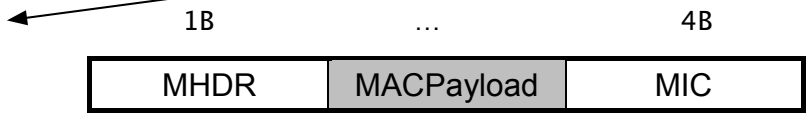


Figure 6: PHY payload structure

MACPayload:

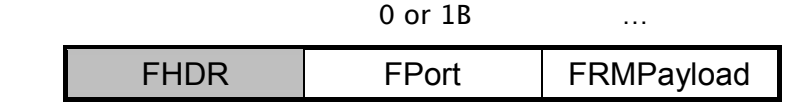


Figure 7: MAC payload structure

FHDR:

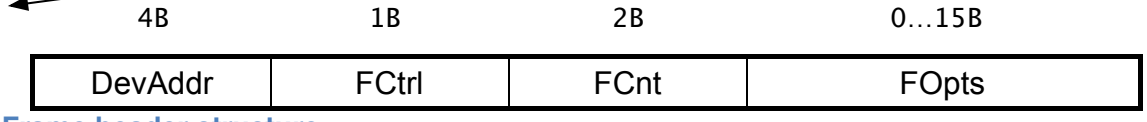


Figure 8: Frame header structure

LoRaWAN network (cont.)

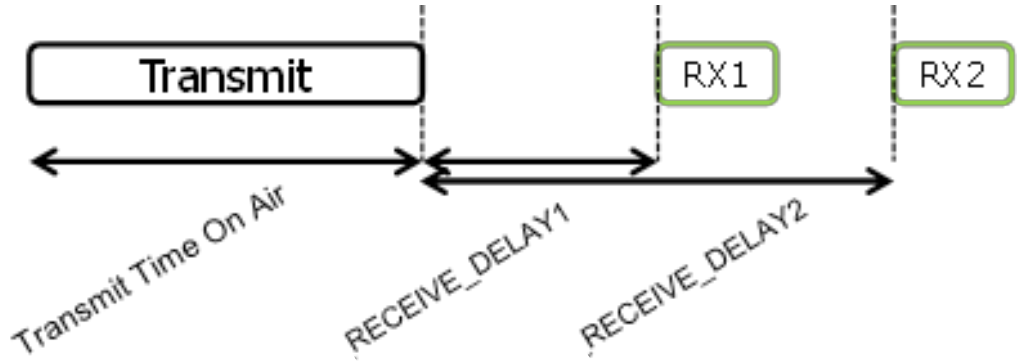
- The frames only carry a single address, the source (or destination) *device* address
- Application demultiplexing : "FPort" (0: pure MAC command)
- *Piggybacking* of MAC commands (power, data rate, channels, device state, rx delay[1] …)
- The network server will typically get several copies of the same frame, received by different GWs (de-duplicated thanks to the frame seq. number)

  The net. server selects the best GW for a reply (if applicable)
- In the core network, frames are forwarded with quite a bit of ancillary data (reception power, timestamp…)

[1] RX2 is always 1 s behind RX1

LoRaWAN Transmissions

- Class A (**A**ll devices)
  - ✓ Exchange **always initiated** by the *device* **Aloha** access
  - ✓ 2 rx windows follow the transmission at +1 s (same channel as TX) and +2 s (channel and SF fixed in advance)



By default : RX2 at 869.525 MHz (center of h1.7), DR0 (SF12, 125 kHz)

  - ✓ Each frame carries the *Confirmed* bit: (expecting and ACK) or *unconfirmed*

- Class B : ***B****eacons* The devices listen periodically to beacons. Regular downlink slots are defined relative to the beacon
- class C : **C**ontinuous reception

Localization

- The observed times of arrival at several GWs allow to compute the differences between them (Time Difference of Arrival TDoA)
- The node is at the intersection of as many ellipses as there are pairs of receivers (4 GWs $\rightarrow$ 6 ellipses)
- Common time reference from GPS at the GWs
- The raw time resolution is $1/BW$… With spread spectrum, spatial resolution is $\# \frac{c}{BW\, 2^{SF}} = 19\,\mathrm{m}$ (9 m @ 250 kHz)
- Precision grows with the number of GWs

A few remarks

- A GW can receive several SF simultaneously ($\approx 30\,dB$ rejection)[2].
- It needs as many reception circuits as there are SFxFreq pairs
  - ✓ 64 packet detectors in parallel (SF5-12 $\times$ 8 channels)
  - ✓ 16 packet receivers
- But no reception while transmitting! (Half duplex)
- Localization is a by product of PHY initial sync.
- Cell breathing
  - ✓ Having more GWs allows to:
    - Lower the SF for closer devices
    - lower the transmission power

[2] C. Goursaud & J.M. Gorce : "Dedicated networks for IoT : PHY / MAC state of the art and challenges"
-8 to -25 dB according to Croce et al.:
'Impact of LoRa Imperfect Orthogonality…"

The Things Network frequency bands EU863-870

Uplink:

- 868.1 - SF7BW125 to SF12BW125
- 868.3 - SF7BW125 to SF12BW125 and SF7BW250
- 868.5 - SF7BW125 to SF12BW125

h1.5

- 867.1 - SF7BW125 to SF12BW125
- 867.3 - SF7BW125 to SF12BW125
- 867.5 - SF7BW125 to SF12BW125
- 867.7 - SF7BW125 to SF12BW125
- 867.9 - SF7BW125 to SF12BW125

h1.4

- 868.8 - FSK — **h1.6**

Downlink:

- **Uplink channels** 1-9 (RX1)
- 869.525 - SF9BW125 (RX2 *downlink only*) — **h1.7**

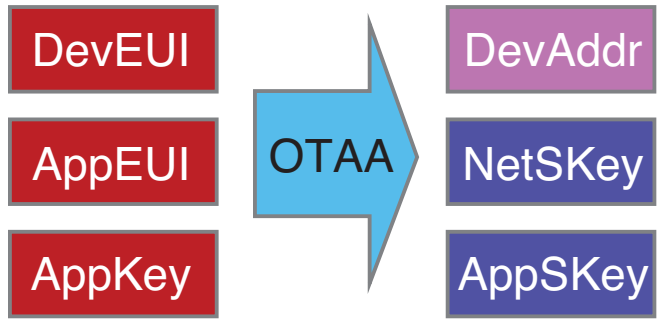
Frame sizes

- Depends on SF : 59B payload at SF12 — SF10, 123 at SF9, 250 at SF8 and SF7…
- 51B @ SF12 $\rightarrow$ 1,3 s of continuous transmission! (and 2.8s for the entire frame)($\Rightarrow$ pause of 4.6 mn before the next transmission in the same band)

Activation

- ABP — *Activation By Personalization*
- OTAA — *Over-The-Air Activation*
  - ✓ DevAddr allocation: the DevAddr is composed of 7 bits of Network ID and then a device-specific addr. (The DevAddr is assigned by the guest network. The real / immutable device identifiers are its NetEUI and AppEUI, which are stored in the device)
  - ✓ Computation of the session keys: AppSKey, NetSKey, from the AppKey (128 bits) stored in the *device*

OTAA — *Over-The-Air Activation (v1.0)*



- Device sends Join Request (AppEUI, DevEUI, DevNonce), signed with
  - ✓ mac=aes128_cmac(AppKey, MHDR | AppEUI | DevEUI | DevNonce)
  - ✓ MIC = mac[0..3]

  (Rx1 receive delay is 5 s for a join request)
- Network server derives session keys:
  - ✓ NwkSKey = aes128_encrypt(AppKey, 0x01 | AppNonce | NetID | DevNonce | pad16)
  - ✓ AppSKey = aes128_encrypt(AppKey, 0x02 | AppNonce | NetID | DevNonce | pad16)

OTAA — *Over-The-Air Activation (v1.0)* (cont.)

- Device authenticates and decrypts Join Accept: AppNonce, NetID, DevAddr
- Device derives (generated at each rejoin):
  - ✓ Network Session Key (NwkSKey), Application Session Key (AppSKey)

Example ABP in TTN

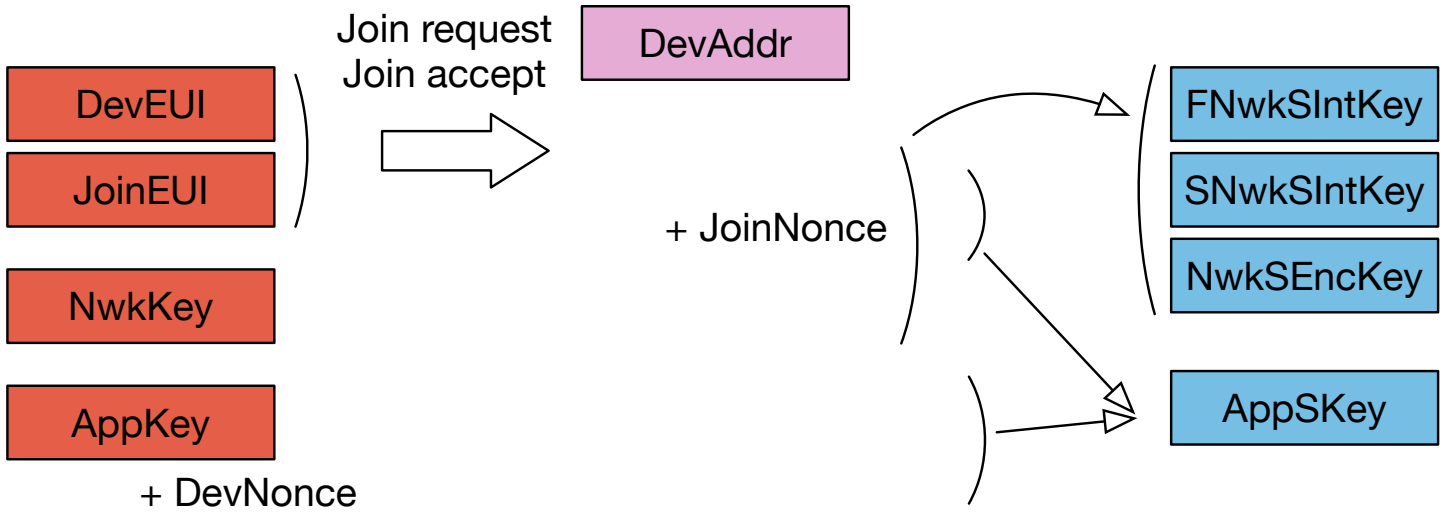
- Register my-device

```
AppEUI=70B3D57EF000002E AppID=hello-world
INFO Generating random DevEUI...
INFO Generating random AppKey...
INFO Registered device
AppEUI=70B3D57EF000002E AppID=hello-world
AppKey=2593946DADA09D86E36E4A4DD3AC632D
DevEUI=00995D884BEBBAB9 DevID=my-device
```

- Personalize my-device

```
INFO Generating random NwkSKey...
INFO Generating random AppSKey...
INFO Requesting DevAddr for device...
INFO Personalized device          AppID=hello-world
AppSKey=48F41F0491B23C804AEB9826162EB9AD
DevAddr=160115EB DevID=my-device
NwkSKey=D7A883537558946974B8DE31D4671617
```

OTAA — *Over-The-Air Activation (v1.1)*



- AppSKey = aes128_encrypt(AppKey, 0x02 | JoinNonce | JoinEUI | DevNonce | pad16)
- FNwkSIntKey = aes128_encrypt(NwkKey, 0x01 | JoinNonce | JoinEUI | DevNonce | pad16)

  SNwkSIntKey = aes128_encrypt(NwkKey, 0x03 | JoinNonce |

  …

  NwkSEncKey = aes128_encrypt(NwkKey, 0x04 | JoinNonce | …

OTAA — *Over-The-Air Activation (v1.1)* (cont.)

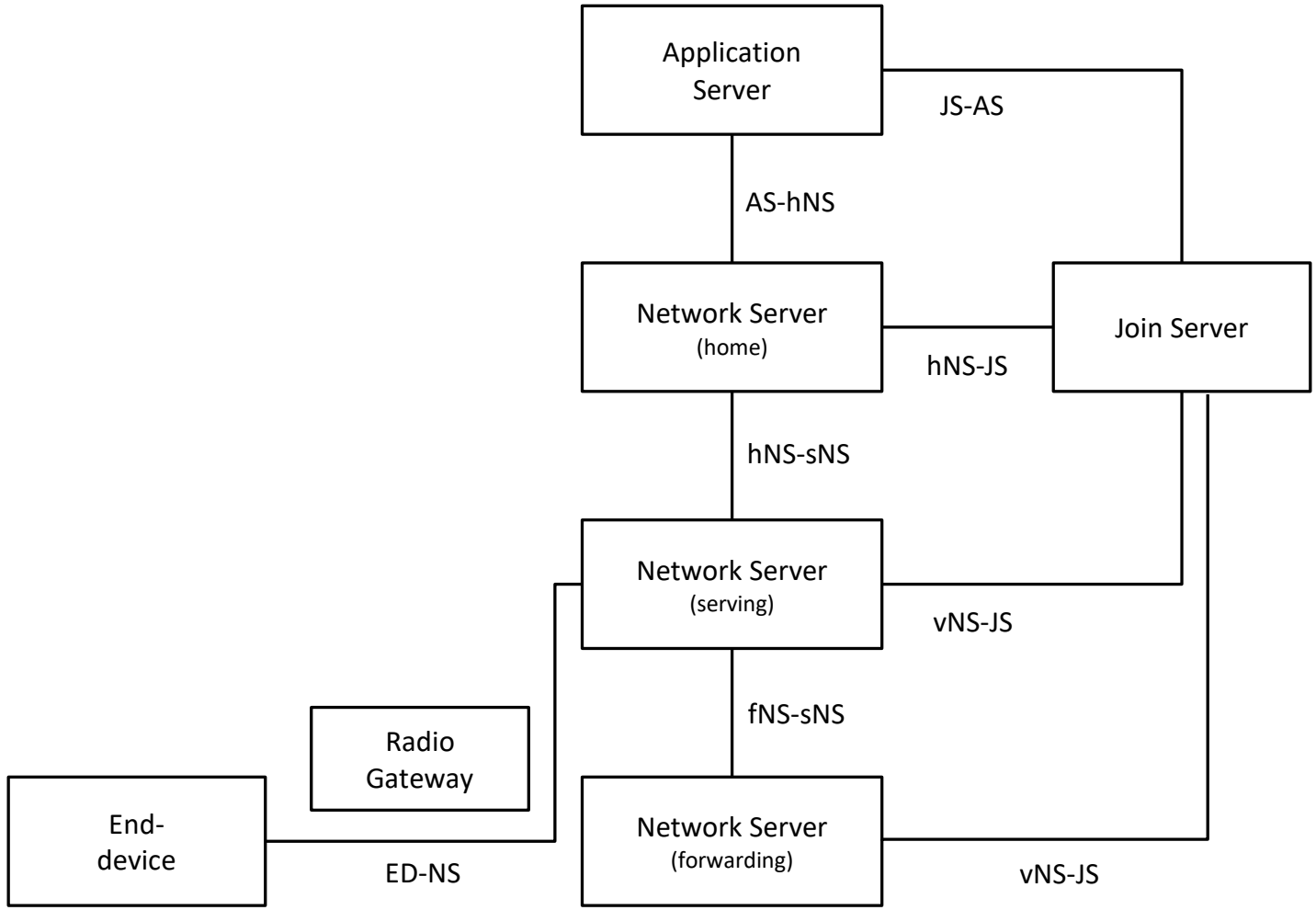
- MIC value of Join accept:
  cmac = aes128_cmac(JSIntKey, JoinReqType | JoinEUI | DevNonce | MHDR | JoinNonce | NetID | DevAddr | DLSettings | RxDelay | CFList)
  MIC = cmac[0..3]

The Network operator has no access to application data

Session keys

- Application session key (AppSKey)
- Forwarding Network session integrity key (FNwkSIntKey)
  for calculating (part of) MIC of UL frames
- Serving Network session integrity key (SNwkSIntKey)
  for computing the MIC of DL messages and half of uplink MIC
- Network session encryption key (NwkSEncKey)
  (en|de)crypt uplink & downlink MAC commands

LoRaWAN Roaming



the forwarding NS only conveys frames between GWs and the serving NS

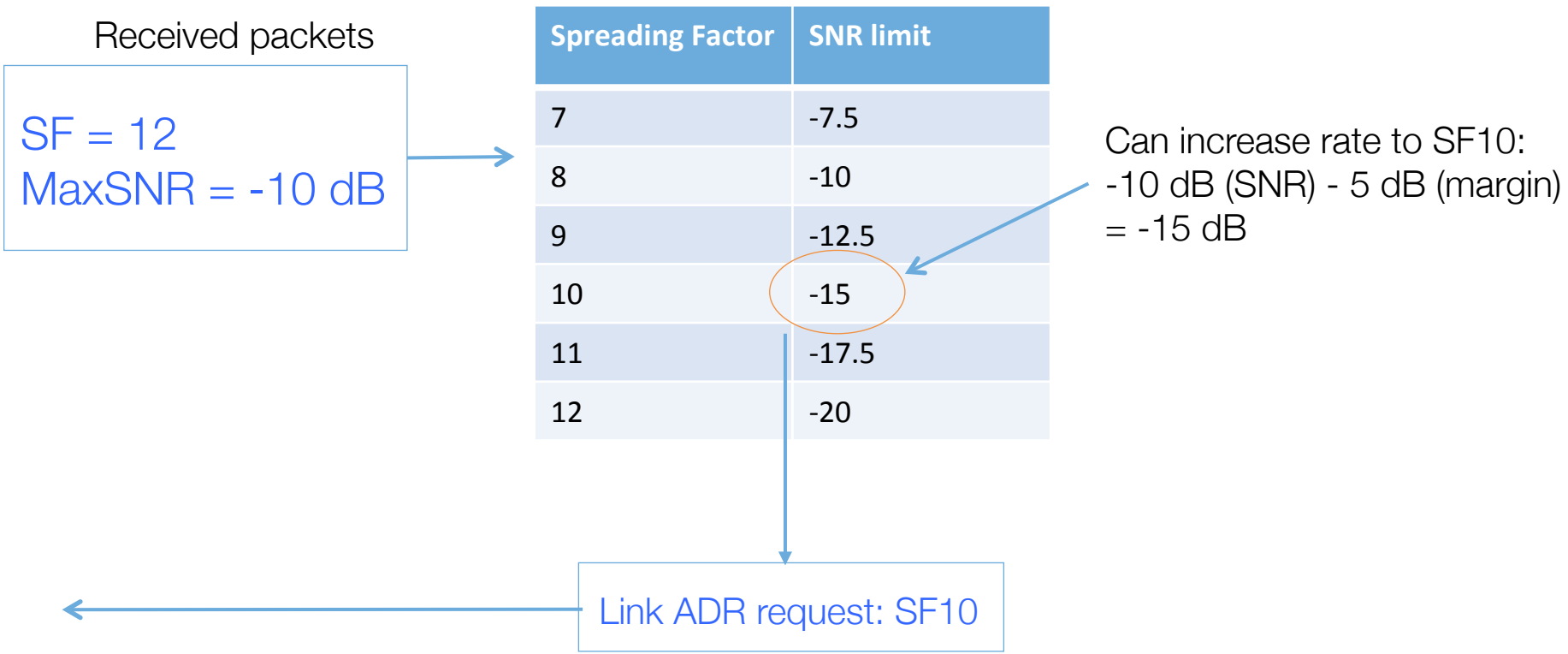
When not roaming, there is only one NS!

MAC Commands

- MAC Commands:
  - ✓ LinkCheckReq, LinkCheckAns: connectivity check
  - ✓ LinkADRReq, LinkADRAns: implement ADR (Adaptive Bit Rate)
  - ✓ DutyCycleReq, DutyCycleAns: update device DC
    RXParamSetupReq, RXParamSetupAns: change RX window
  - ✓ DevStatusReq, DevStatusAns: get dev. status like battery
  - ✓ NewChannelReq, NewChannelAns: channel update
  - ✓ RXTimingSetupReq, RXTimingSetupAns: change RX window
  - ✓ (v1.1) RekeyInd, RekeyConf: Rekey commands
- Set power, SF, data rate, channels, device state, RX delay[3]
- FPort: kind of Application ID (0: pure MAC command to/from Network Server, otherwise: pass to the application layer)
- Commands sent in a separate data frame in the FRMPayload (FPort field set to 0) or piggybacked in FOpts

[3] RX2 is always 1 s behind RX1

ADR



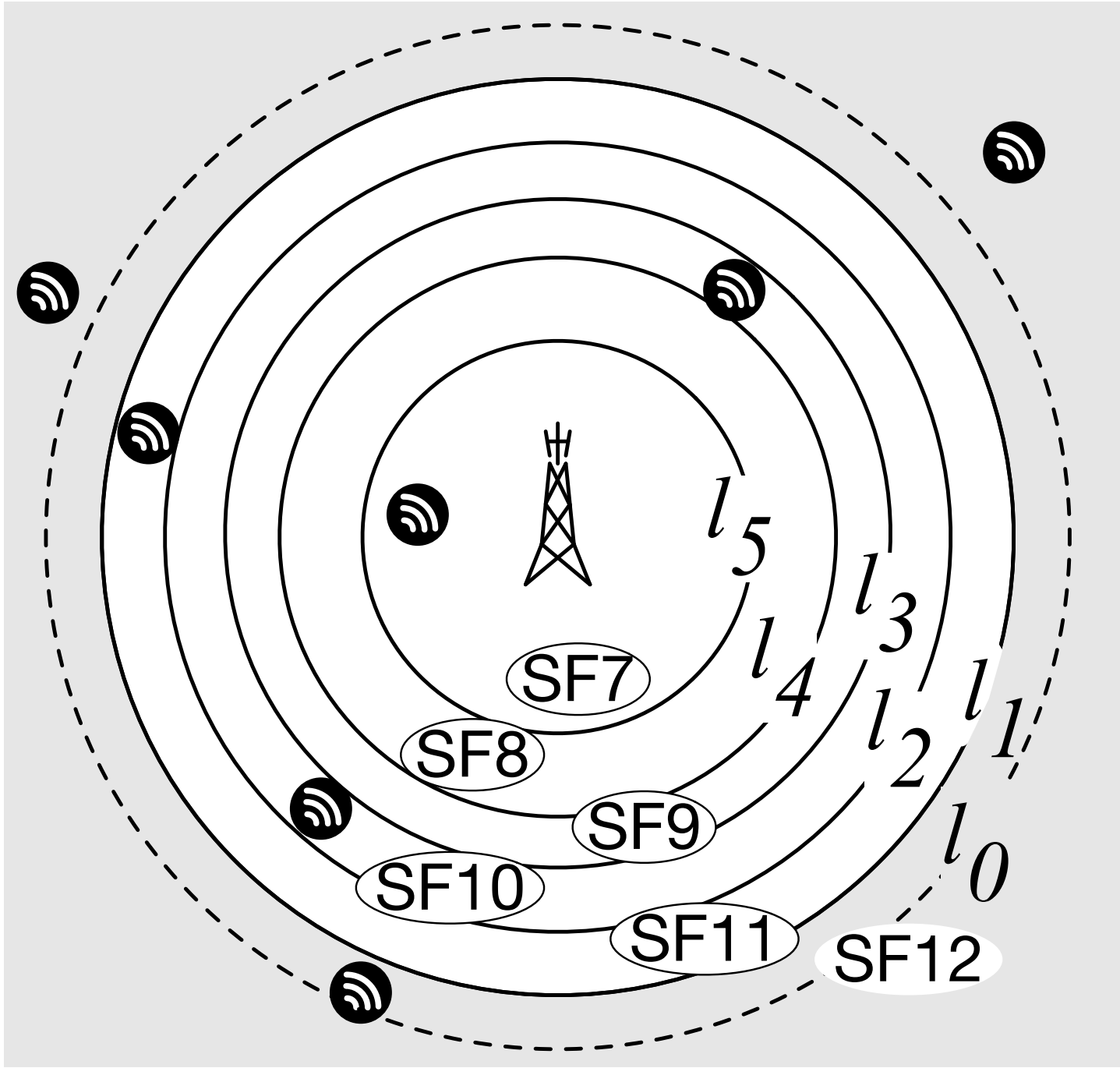
- If ADR bit is set, the network controls data rate through MAC commands
- Network server estimates SNR of last 20 packets (long!), chooses suitable data rate, SF, and TP
- Sends the parameters in LinkADRReq

ADR (cont.)

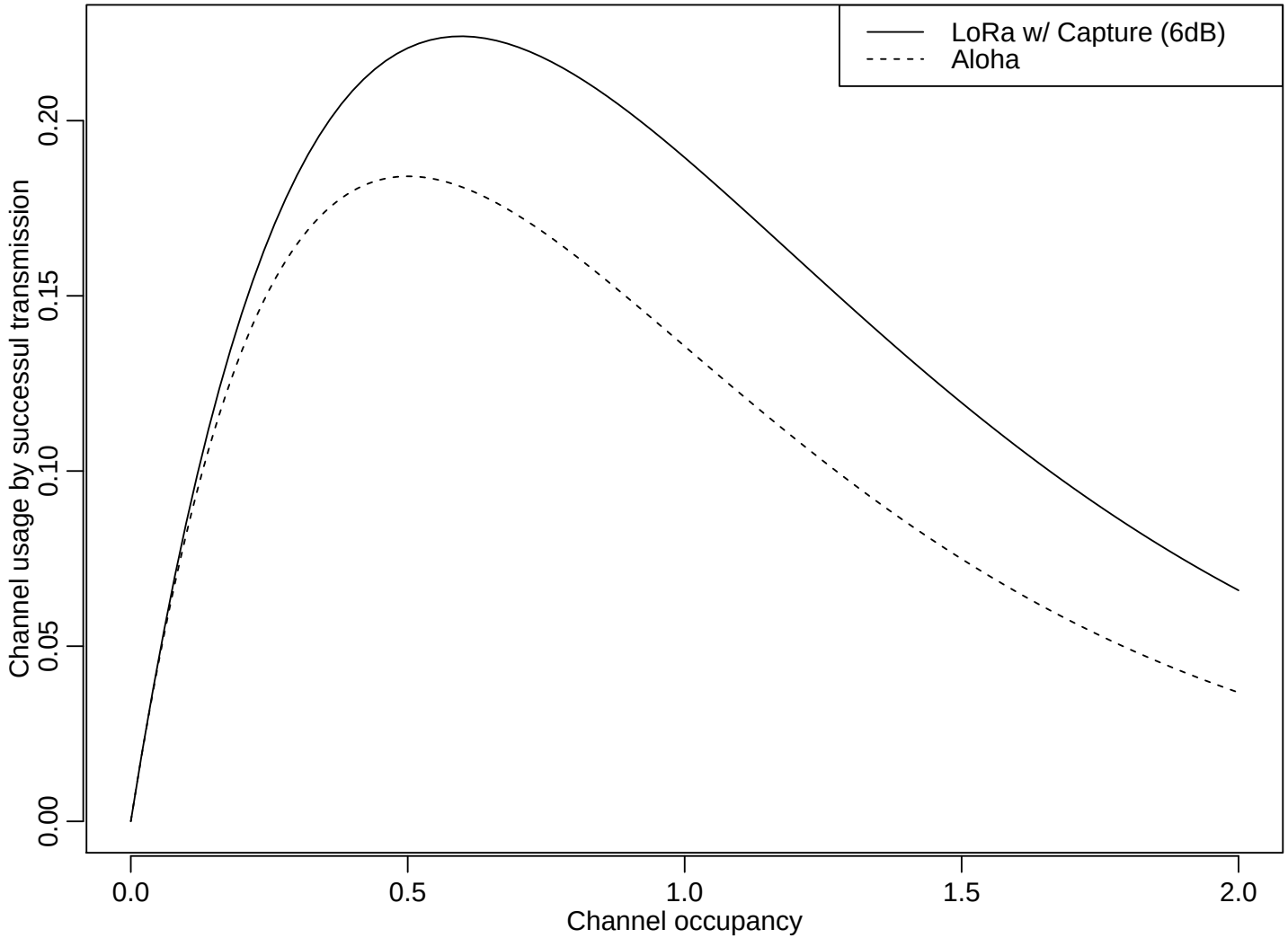
- After ADR_ACK_LIMIT frames with no downlink response, the devices asks to hear from the network by setting the ADRACKReq bit in the frame header.
- The network then has ADR_ACK_DELAY frames to reply

Connectivity problem?
Try to regain connectivity!!

- Each time ADR_ACK_DELAY frames pass with no answer, first increment Tx power. When max power is reached, then increment SF.


l_5
l_4
l_3
l_2
l_1
l_0
SF7
SF8
SF9
SF10
SF11
SF12

LoRa: Aloha access with physical capture



Lora transmissions over a Rayleigh channel, short range: Random gain variations often allow to capture one of the colliding frames

Example of SF allocation vs range