

# Communication radio robot/sol ou multi-robots

Matthieu Herrb



Journée <sup>2</sup>RM 4 juillet 2018

## Motivations :

- Source de problèmes sans fin dans expérimentations réelles,
- Expériences de liaisons radio sur toutes sortes de robots depuis près de 20 ans
- [Tetaneutral.net](#) & FFDN,..
- Essayer d'explorer/développer des solutions pratiques plutôt que de faire de la recherche sur le sujet.

# Spécifications

- Environnement mono ou multi-robots de taille humaine (quelques dizaines de mètres à plusieurs kilomètres)
- Communications radio « standard » (bandes libres, tél. mobile; à l'exclusion de bandes soumises à licences)
- Perturbations (obstacles,...)
- Robots qui peuvent tomber en panne
- Télé-opération / debug / autonome
- Débits / latence
- Interférences ; gestion des fréquences



# Le cas de TCP

*Wikipedia* : « Transmission Control Protocol (littéralement, « protocole de contrôle de transmissions »), abrégé TCP, est un protocole de transport fiable, en mode connecté, documenté dans la RFC 7931 de l'IETF. »

## Mais :

- timeouts très longs (plusieurs minutes)
  - difficile de détecter une perte de connexion
  - retards dans la transmission des données
- reprise en cas de déconnexion lourde
- algorithmes de contrôle de congestion peu adaptés pour environnement wifi dynamique
- *bufferbloat*

À éviter... (mais reste largement utilisé en robotique pour communications radio)

- Ne pas utiliser de sockets TCP directement  
→ isoler les communications « externes »  
dans des composants dédiés
- Privilégier UDP
- Réseaux mesh / ad-hoc
- Communication multi-modale (Wifi/3G/...)
- Monitorer la qualité du signal / de la communication
- Prendre en compte la communication au niveau de la navigation / supervision

Exemple : projet COMETS (2003-2005)  
coopération entre UAVs « lourds » et station centrale :

- Middleware spécifique : **BBCS**
- Tableau noir partagé entre les robots et station centrale
- Utilisable sur plusieurs supports : liaison série, UDP, TCP
- Contrôle explicite de la bande passante pour éviter saturation
- Retours sur la qualité des communications
- Au départ architecture type blackboard  
→ revue pour permettre supervision
- Pas libre (et probablement plus maintenu) :-)

## Autre exemple : DDS

- standard OMG
- utilisé dans ROS2
- notions de qualité de service / priorités
- plusieurs implémentations
  - libres
  - commerciales

# Composants dédiés

- Isole les composants internes
  - pas de blocage
  - pas besoin de dupliquer code de reprise sur erreurs
- Centralise les problématiques
  - qualité des liens
  - nommage/adressage
  - pertes de communications
  - reconfiguration

Exemples : **Martha** (1997-2000) : CI, IRC, **urus** (2006-2009)



Nombreux résultats théoriques (simulation)

Exemple de résultat pratique : RT-WMP

- développé au CUD de Saragosse
- contexte : collaboration multi-robots pour défense civile
- protocole à passage de jeton
- utilise le mode Ad-Hoc de IEEE-802.11
  - temps borné de bout en bout
  - supporte plusieurs HOP
  - gestion de priorités



# Communication multi-modale

Nombreux media de communication disponibles :

- Wifi 2.4GHz / 5 GHz
- Liens radios série (866MHz)
- Zigbee, Bluetooth & co
- 3G / 4G / LTE
- câbles

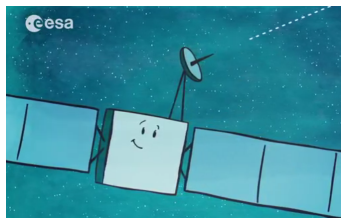
Plusieurs approches possibles :

- Explicite : changements de mode décidés par opérateur
- Implicite : transparente

Dans tous les cas : politique de choix du mode...

# Prise en compte de la qualité de communication

- Filtrage des données des dispositifs radio
- Cartographie du niveau de communication
- Contraintes externes (rendez-vous avec relais)
- Planification des actions de communication
- Déformations des trajectoires pour garantir communications
- Actions de relais de communication



- Authentification mutuelle robot / robot ; robot / station
- Chiffrement des données transmises
- Protection contre déni de service
- Détection des attaques ; remontée d'alertes

« Y'à qu'à »

# Contraintes de la vraie vie

Communication « shell » avec un robot...

TCP mais outils :

- tmux (screen)
- mosh
- NX / x2x / x2go → sessions X windows

Partage de la bande passante avec données opérationnelles ?

Problème :

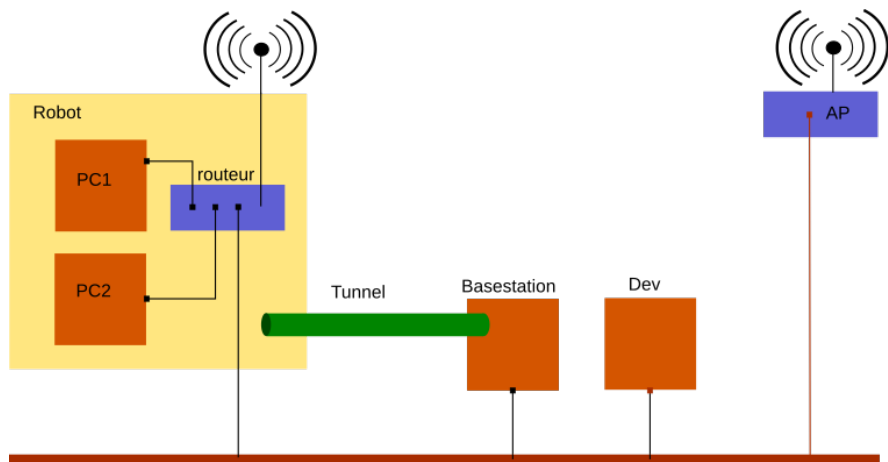
- robot avec plusieurs calculateurs à bord
- connexion wifi ou filaire transparente
- adresses IP sur le LAN des postes de travail

Solution (inspirée par Willow Garage):

- routeur Wifi en mode client sur le robot
- tunnel UDP entre robot et une station de base
- le robot surveille l'état des connexions physiques
- route le trafic sur le meilleur support

<https://git.openrobots.org/projects/roralink>

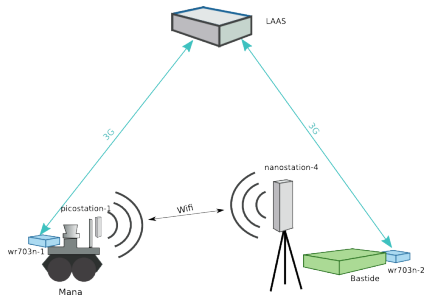
# Roralink



# Robobox : Un relais multi-mode transparent

Extension de roralink : tout dans le routeur (OpenWRT)

- possibilités : filaire / Wifi (plusieurs bandes) / 4G
- interface de contrôle / monitoring (ROS ou API générique)
- cloud privé pour router communications 3/4 G





# Conclusion

- Améliorer les pratiques
  - prise en compte explicite contraintes radio (QoS)
  - séparer protocoles réseau et application
- Standardiser des outils
- Suivre les progrès des technologies radio / IoT / M2M

Questions ?