

Actas de las Jornadas Uruguayas de Ciencias de la Computación 2024 “Jorge Vidart”

Instituto de Computación, Facultad de Ingeniería
Universidad de la República, Uruguay



Editores

Sergio Nesmachnow
Pedro Moreno-Bernal



UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY



Implementación y estudio de una maqueta de red 5G SA basada en srsRAN

Facundo Pedreira^[0009–0003–2959–9985], Bruno Tió^[0009–0005–1077–1604],
Leonardo Barboni^[0000–0002–7036–4083], and Claudina
Rattaro^[0000–0001–7149–5934]

Facultad de Ingeniería, Universidad de la República,
Montevideo, Uruguay
{facundo.pedreira,brunot,lbarboni,crattaro}@fing.edu.uy

Abstract. En un contexto donde las tecnologías de redes móviles avanzan y cambian constantemente, resulta de gran interés contar con maquetas dedicadas a la experimentación e investigación en esta área. Este proyecto tiene como fin la implementación de un *testbed* de red 5G *Stand Alone* de extremo a extremo, basado en el software de código abierto *srsRAN Project*, utilizando equipos de radio definida por software como terminales de radiofrecuencia. Se propone además realizar una comparativa de *performance* con su principal competidor *OpenAirInterface*, con el objetivo de determinar, a día de hoy, cual de estas opciones resulta más útil como base para futuros desarrollos. Los resultados preliminares señalan que *OpenAirInterface* ofrece un rendimiento superior en términos relativos, mientras que *srsRAN Project* se destaca por su simplicidad y el reducido tiempo requerido para su puesta en marcha.

Keywords: 5G NR · Red de Acceso Móvil · Software Defined Radio · srsRAN

1 Introducción

El mundo de las redes móviles está en constante evolución y el despliegue de redes de quinta generación (5G) viene siendo el eje central en los últimos años a nivel mundial. Con las comunicaciones mejoradas de banda ancha móvil (eMBB), 5G soporta altas velocidades de transmisión de datos para aplicaciones como la realidad virtual y aumentada. Las comunicaciones ultra fiables y de baja latencia (URLLC) abren nuevas posibilidades para servicios críticos, incluyendo vehículos autónomos y telemedicina, mientras que las comunicaciones masivas tipo máquina (mMTC) facilitan el despliegue de ecosistemas IoT (Internet de las Cosas) con miles de millones de dispositivos conectados. Para garantizar el cumplimiento de estos casos de uso aún existen varios desafíos tecnológicos que sin dudas se pueden extrapolar a lo que serán las redes de sexta generación, entre ellos están: la gestión eficiente e inteligente de los recursos para el cumplimiento

2 F. Pedreira, B. Tió, L. Barboni and C. Rattaro

de la calidad de servicio (QoS), despliegue de *Network Slicing* en los distintos segmentos de la red, interoperabilidad entre sistemas heterogéneos, abordaje de problemas de ciberseguridad, etc [2].

La evolución de las redes móviles ha estado históricamente marcada por *hardware* de propósito específico, con funciones de red estrechamente vinculadas a los fabricantes, lo que limitaba tanto la flexibilidad como la accesibilidad. Sin embargo, tecnologías como el Radio Definida por Software (SDR) [5] han transformado este panorama, permitiendo que las implementaciones de redes inalámbricas, y en particular las redes móviles, sean más accesibles para la investigación y la educación. La SDR es un sistema de comunicaciones donde las funciones que tradicionalmente se implementaban en *hardware* ahora se realizan mediante *software*, utilizando un ordenador personal común y un dispositivo de radiofrecuencia genérico, de bajo costo y conectado por USB. Esto permite crear equipos de comunicación capaces de interactuar con diversos sistemas, como televisión, radio AM y FM, e incluso aplicaciones específicas como estaciones base o equipos de usuario en redes móviles.

A pesar de estos avances, muchas de las soluciones disponibles para realizar pruebas realistas en redes móviles aún dependen de *hardware* costoso y *software* propietario, lo que representa una barrera significativa para equipos con recursos limitados. Como respuesta, han emergido alternativas de código abierto, como *srsRAN*[9] y *OpenAirInterface*[6], que ofrecen herramientas robustas para desarrollos locales y autónomos. Estas soluciones democratizan el acceso a las tecnologías de redes móviles, fomentando la innovación y abriendo nuevas posibilidades para la investigación y la educación en este campo.

srsRAN y *OpenAirInterface* buscan ofrecer una implementación flexible de tecnologías como LTE y 5G NR. Por un lado *OpenAirInterface* es gestionado por la OpenAirInterface Software Alliance (OSA), un grupo internacional de empresas, universidades e instituciones, incluyendo grandes actores de telecomunicaciones como Nokia, Vodafone y Ericsson. Este grupo colabora regularmente con el 3rd Generation Partnership Project (3GPP), la organización que produce las especificaciones técnicas que impulsan las diferentes generaciones de tecnologías de telecomunicaciones.

Por otro lado, el proyecto *srsRAN* es desarrollado por la compañía Software Radio Systems (SRS), una empresa privada que provee tanto el proyecto de código abierto *srsRAN* como una versión paga ofrecida a empresas junto con soporte continuo. La implementación *open-source* de *srsRAN* 5G es el segundo proyecto de este tipo implementado por la compañía, ya que esta ya cuenta con una versión similar para redes 4G.

La principal diferencia entre ambos radica en que *OpenAirInterface* cubre tanto la red de acceso por radio como el núcleo de red, mientras que *srsRAN Project* usa un núcleo externo, *Open5GS*, lo que le permite ofrecer una mayor modularidad y menor consumo de recursos. Los trabajos previos que comparan estas soluciones se han centrado en su mayoría en sus versiones LTE. En contraste, el estudio más reciente relacionado con este tema [10], publicado este año, es uno de los pocos que aborda desarrollos basados en 5G. En este trabajo,

los autores montaron su maqueta utilizando dos SDR USRP Ettus B210, pero para las comparativas optaron por dispositivos comerciales (COTS UE), en lugar de emplear las implementaciones de UE ofrecidas por *OpenAirInterface* y *srsRAN Project*. Aunque este estudio representa un avance al explorar las versiones más recientes de 5G, no explota completamente todas las posibilidades de comparación disponibles, dejando espacio para análisis más exhaustivos en el futuro.

En esta línea, el presente trabajo implementa un *testbed* basado en *srsRAN Project* y realiza una comparación directa con un estudio previo que analizó en detalle *OpenAirInterface* [7], utilizando el mismo *hardware* de base. Este enfoque garantiza una evaluación equilibrada de ambas soluciones en un entorno uniforme, lo que permite destacar sus fortalezas y debilidades de manera justa. Los resultados preliminares indican que *OpenAirInterface* muestra un mejor desempeño relativo en términos de rendimiento, mientras que *srsRAN Project* sobresale por su facilidad de uso y el tiempo reducido necesario para su implementación.

El resto del artículo se estructura de la siguiente forma. En la sección 2 se realiza una comparativa del estado actual de las dos soluciones. La sección 3 es la medular del artículo donde se describe en detalle la maqueta implementada, el *hardware* y *software* utilizado y las versiones de los distintos componentes. En la sección 4 se presentan y analizan los principales resultados obtenidos. Finalmente en la sección 5 se concluye el trabajo y se indican líneas de trabajo en curso y futuro.

2 La solución srsRAN Project y su comparativa con OpenAirInterface

En esta sección se presenta una comparativa entre las dos soluciones *open-source* más destacadas para el despliegue de redes 5G: *srsRAN Project* y *OpenAirInterface*. La comparación se centra en las características más relevantes, con especial énfasis en las diferencias funcionales que cada solución ofrece. Un resumen de las características más importantes evaluadas se puede ver en la tabla 1.

Entre los puntos en común, se destaca que ambas soluciones soportan la implementación de estaciones base (conocida como gNB) en modos *Standalone* (SA) y *Non-Standalone* (NSA). Asimismo, ambas pueden operar en un amplio rango de anchos de banda, *Subcarrier Spacing* variable y soportan *duplexing* en *Frequency Division Duplexing (FDD)* o *Time Division Duplexing (TDD)*, de acuerdo con lo estipulado en el estándar desarrollado por la 3GPP[1]. Sin embargo, es destacable que *OpenAirInterface* alcanza valores más altos de *Subcarrier Spacing*, lo cual teóricamente permite latencias menores y la posibilidad de operar a frecuencias más altas. A su vez, permite trabajar con *Dynamic TDD*, que permite asignar y reasignar de forma dinámica los recursos de tiempo entre *Uplink* y *Downlink*.

A su vez, existen diferencias significativas que resultaron clave durante la implementación y evaluación del desempeño. A diferencia de *OpenAirInterface*, *srsRAN Project* no ofrece una implementación para un UE (equipo de usuario)

Table 1: Comparativa entre srsRAN y OpenAirInterface

	srsRAN Project	OpenAirInterface
Generalidades		
Versión actual	24.10	2024.w49
gNB SA y NSA	Sí	Sí
UE SA y NSA	No. Utiliza srsUE 4G	Sí
Core Network	No. Utiliza Open5GS	Sí
RAN		
Subcarrier spacings	15/30 kHz*	15 kHz, 30 kHz, 60 kHz, 120 kHz
Bandwidths	5/10/15/20/25/30/40/50/60/70/80/90/100 MHz*	10/20/40/60/80/100 MHz
Duplexing	FDD, TDD*	FDD, TDD, dynamic TDD
Split CU/DU	Sí, oficial	Sí, desarrollado por comunidad

*Si bien *srsRAN* soporta SCS de 30 kHz, TDD y BW mayores a 20 MHz, estos no pueden ser usados por *srsUE* 4G.

nativo de 5G. En su lugar, utiliza *srsUE* en su versión 4G adaptada, lo que limita su funcionalidad en el contexto de redes SA 5G. Este aspecto no es menor, ya que, como se analizará más adelante, restringe las capacidades operativas de la maqueta. Según lo declarado en el *roadmap* del proyecto *srsRAN*, no se planea el lanzamiento de un UE 5G nativo en el futuro cercano [9]. Asimismo, tampoco proporciona una solución propia de *Core* 5G, dependiendo de software de terceros como *Open5GS*, pudiendo esta desplegarse en *bare-metal*, sobre un sistema Unix o incluso en un contenedor *Docker*.

Por último, cabe destacar que *srsRAN Project* incluye una implementación oficial de un *split CU/DU* y promete una próxima implementación de una separación *CU-UP/CU-CP*. Este desarrollo resulta particularmente atractivo para futuras implementaciones de menor escala, como se discutirá en la sección 5.

3 Implementación

La maqueta de red implementada se compone de los dos elementos principales: núcleo (5GCN) y red de acceso (NR-RAN); esta última modelada por sus equipos principales (gNB) y equipo de usuario (UE) según se muestran en la figura 1. En cuanto a *software*, como *srsRAN Project* carece de un núcleo de red propio, se usa como opción *Open5GS* en un contenedor *Docker*, y junto a este los módulos *srsNB* y *srsUE* para implementar la estación base y el equipo de usuario, respectivamente. A continuación se dan más detalles de los equipos involucrados en el despliegue así como también los detalles de ejecución.

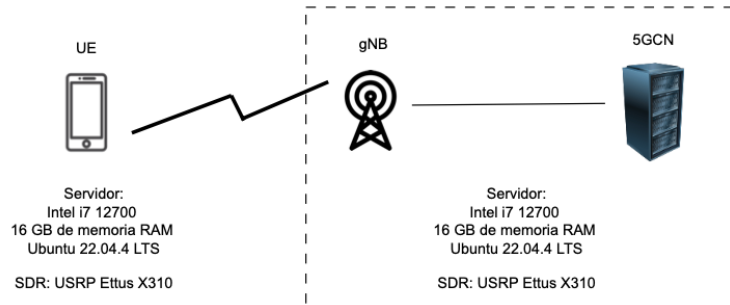


Fig. 1: Elementos de la maqueta indicando sus características de *hardware* principales.

3.1 Equipamiento utilizado

El *hardware* de base es idéntico al del trabajo previo [7], destacándose que para implementar la interfaz de aire se utilizan equipos de Software Define Radio (SDR) de alto porte soportando todo el rango de frecuencias definido como FR1 en 5G-NR [1]. En la figura 2 se muestran dos imágenes que reflejan los equipos utilizados, destacándose los SDR, antenas y computadoras.

Las computadoras utilizadas cuentan con procesadores i7 14700K (con 20 núcleos de procesamiento) y 16 GB de memoria RAM, lo que se considera más que suficiente para ejecutar el software utilizado en la maqueta, así como programas adicionales utilizados para monitorear el funcionamiento de la misma. De esta manera, se aseguró que los recursos de *hardware* no fueran una limitante al momento de medir el rendimiento de la solución implementada.

El otro equipo de *hardware* fundamental en esta implementación fue el SDR USRP Ettus X310. Este tipo de equipos se utilizan para emular en software los diferentes componentes de *hardware* que componen una antena especializada, como filtros, amplificadores y moduladores. De esta manera, se simula el comportamiento de las antenas especializadas necesarias para implementar una radiobase sin contar con el hardware específico para las mismas. En este caso, el SDR X310 cuenta con *daughterboards* UBX de banda ancha con hasta 120 MHz de ancho de banda, cubriendo entonces todo el rango de frecuencias y anchos de banda soportados por *srsRAN Project*. A su vez cuenta con una placa FPGA programable de alta velocidad y una interfaz de 10 Gbps a la PC, lo que permite nuevamente utilizar el SDR sin que la capacidad de procesamiento sea una limitante que actúe sobre el rendimiento que se busca medir en la maqueta [3].

Además de las computadoras, equipos SDR, conectores, atenuadores, antenas; se utilizó un teléfono comercial tal como se detallará más adelante.

6 F. Pedreira, B. Tió, L. Barboni and C. Rattaro

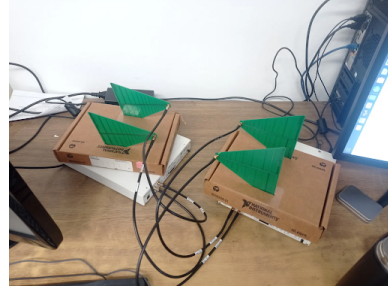
3.2 Puesta en operación

Una vez instalados todos los componentes de software y realizada una configuración básica, en primera instancia se buscó lograr una conexión estable entre la gNB y el UE por la interfaz de aire. Para esto se ajustaron de forma muy precisa las ganancias tanto en transmisión como en recepción; resultando óptimos los valores de 10 dB en transmisión y 5 dB en recepción. Para verificar esta estabilidad y el comportamiento coherente de la conexión se realizaron pruebas iniciales en las que se variaron diferentes parámetros como:

- Anchos de banda: 5, 10, 15 y 20 Mhz
- Bandas: n1, n2, n3, n5, n7 y n8
- Distancias entre transmisor y receptor: 0.3 m, 1 m y 3 m



(a) Maqueta completa. Incluye servidores, SDRs y antenas



(b) Equipos SDR transmitiendo a muy corta distancia

Fig. 2: Disposición de equipos utilizados en la maqueta completa.

La metodología implementada para estas pruebas iniciales fue la siguiente:

1. Establecer la configuración recomendada en la web de *srsRAN Project* [8].
2. Confirmar la conexión con la configuración recomendada.
3. Variar uno de los parámetros y verificar conectividad nuevamente.
4. Generar tráfico artificial que sature la red por 180 segundos verificando conectividad durante todo el test.

Una vez que la maqueta cumplió exitosamente con estas pruebas, se pasó a la etapa de evaluación. Para ello se tomó como medida de desempeño el *Throughput* alcanzado en Mbps. Si bien este no es el único parámetro para evaluar la performance (también es posible analizar sus tiempos de latencia, tasas de pérdida o conexiones simultáneas permitidas, entre otros), gran parte del valor de este trabajo radica en la comparación con *OpenAirInterface*, y dado que el trabajo con el cual se realizó la comparativa [7] realizó pruebas de capacidad, son estos los valores con los que se puede contrastar.

4 Principales Resultados

Se tomaron un conjunto de *setups* fijos que permiten medir los diferentes factores que afectan al rendimiento de la red, aislando y modificando las correspondientes variables en la configuración. Fijando el *hardware* utilizado y variando exclusivamente parámetros de configuración, se pudo registrar mediante la herramienta *iperf3* las velocidades y latencias obtenidas al variar valores como la banda y frecuencia de transmisión, el ancho de banda y frecuencia de muestreo utilizados, e incluso la distancia física entre antenas. Esta aplicación es del tipo cliente-servidor, y permite generar tráfico artificial para saturar la red y medir su capacidad durante un intervalo de tiempo deseado.

A su vez, se monitoreó el uso de recursos de los procesos correspondientes al núcleo de red y la gNB durante estas pruebas mediante el comando *htop* de linux, con el fin de evaluar así el rendimiento en el hardware utilizado. Este comando permite listar en tiempo real el uso de diferentes recursos de hardware de cada proceso ejecutando en el sistema:

- Uso de núcleos del procesador
- Memoria virtual solicitada
- Memoria RAM asignada
- Memoria de swap

4.1 Medidas de Throughput

Se presentan entonces los resultados obtenidos al variar los diferentes anchos de banda, los cuales resultan de interés para esta instancia. Durante estas pruebas se trabajó con los siguientes parámetros:

- Banda: n3
- Ganancias: Tx 10dB, Rx 5dB
- SubCarrier Spacing (Δf): 15kHz
- Distancia: 1m

En la gráfica de la figura 3 se presentan los resultados obtenidos, agrupados según la cantidad de PRB utilizados. Cabe recordar que los *Physical Resource Blocks* (PRB) constituyen las unidades fundamentales de recursos en el dominio de la frecuencia, asignadas a los usuarios para la transmisión de datos. Cada PRB está compuesto por 12 subportadoras contiguas, cuya separación está determinada por el valor del *SubCarrier Spacing* (Δf). La ecuación (1) muestra cómo se calcula la cantidad de PRB disponibles.

$$N_{PRB} = \left\lfloor \frac{BW_{usable}}{12 \times SCS} \right\rfloor \quad (1)$$

Al comparar estos resultados con las pruebas realizadas en [7] se pueden extraer varias observaciones preliminares.

En primer lugar, se constató que las pruebas de aire en [7] se llevaron a cabo en las bandas n78 y n41, ambas operando en TDD. Sin embargo, como

8 F. Pedreira, B. Tió, L. Barboni and C. Rattaro

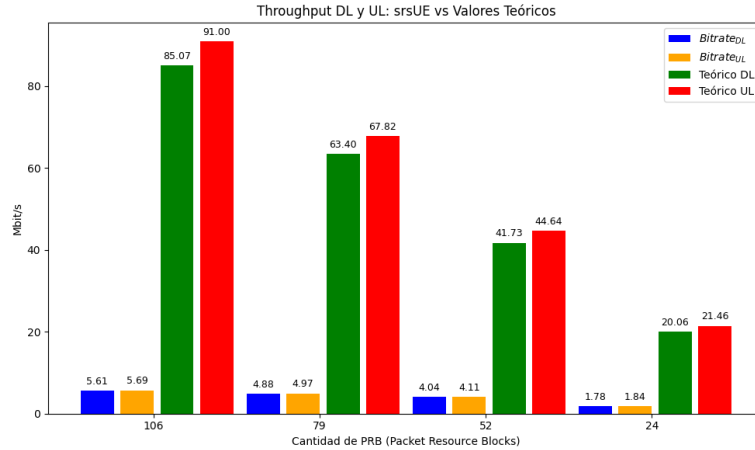


Fig. 3: Throughputs alcanzados por srsUE comparados con valores teóricos, agrupados por cantidad de PRB

se destacó en la sección anterior, el *srsUE* utilizado corresponde a una versión modificada de la implementación 4G, adaptada para establecer la conexión. Al no contar con todas las capacidades de un UE nativo 5G, esta versión carece de la funcionalidad necesaria para operar en TDD. Además, se observó que el trabajo previo utilizó un espaciado entre subportadoras de $\Delta f = 30$ kHz. Esto tampoco pudo ser replicado, ya que *srsUE* únicamente soporta un espaciado de $\Delta f = 15$ kHz.

Estas diferencias resultan de suma importancia, ya que imposibilitan una comparación directa de los resultados de *throughput*, dado que las configuraciones presentan capacidades distintas en los canales de comunicación establecidos. Por este motivo, se decidió adoptar como métrica el rendimiento de cada maqueta respecto al valor teórico esperado, expresado en forma porcentual. Esta metodología permite evaluar el grado de eficiencia alcanzado por cada implementación en relación con su máximo potencial, proporcionando así una base objetiva y equitativa, independientemente de las diferencias en configuraciones y capacidades de las soluciones. En la gráfica 4, se presentan los porcentajes obtenidos para cada solución, tanto en *Downlink* como en *Uplink*, agrupados según la cantidad de PRB utilizados.

Comparando los resultados obtenidos, se concluye que ambas implementaciones alcanzan rendimientos que permiten operar la maqueta en las configuraciones deseadas, garantizando la conectividad de extremo a extremo. No obstante, la implementación de *OpenAirInterface* logra valores de *bitrate* superiores en relación con el valor teórico esperado. Por otro lado, los valores obtenidos con la maqueta implementada usando *srsUE* se consideran extremadamente bajos, ya que en ningún caso superaron el 10% del valor teórico.

Implementación y estudio de una maqueta de red 5G SA basada en srsRAN 9

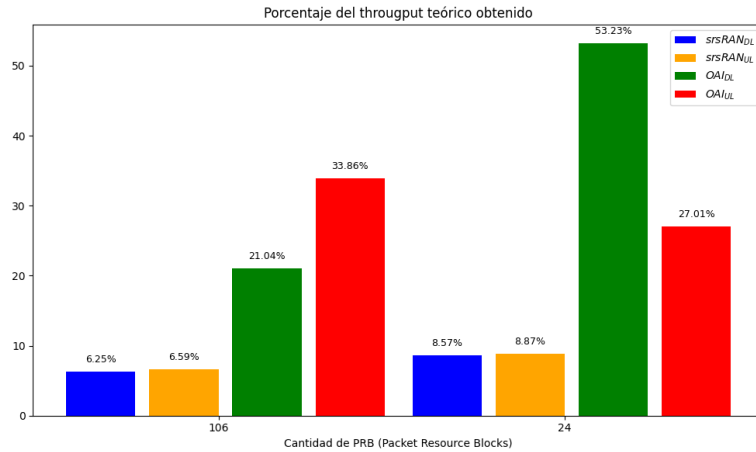


Fig. 4: Valores porcentuales del throughput alcanzado por cada solución respecto al valor teórico.

Llegado este punto, lo siguiente fue cuestionar dónde podía estar el punto de congestión del sistema. Naturalmente, el primer sospechoso fue el srsUE 4G. Esta implementación se encuentra limitada a configuraciones que, si bien logran conectividad a una gNB 5G, son inherentes a tecnología previa. De esta forma, no estaría pudiendo hacer uso de los protocolos e implementaciones que trae esta nueva generación.

Para confirmar esta tesis, se propuso conectar a la radiobase un dispositivo COTS UE (*Commercial Off The Shelf User Equipment*). En este caso, un dispositivo celular que cuenta con conectividad 5G nativa, como es el caso de un *Samsung Galaxy A22 5G*. Para esto se debió contar con una grabadora de tarjetas SIM y las respectivas tarjetas programables. Luego de configurarlo debidamente, se logró conectarlos a la red de acceso. Así, se repitieron las pruebas originales para diferentes anchos de banda, pero esta vez usando el mencionado dispositivo como UE. Los resultados pueden observarse en la gráfica de la figura 5.

Como se puede observar, el salto en la capacidad de tráfico fue abismal. El COTS UE alcanzó un bitrate de aproximadamente 80% del valor teórico en la mayoría de los casos. Esta diferencia, si bien determina fehacientemente que sea srsUE 4G el culpable de los bitrate bajos, indica claramente que el núcleo y la radiobase implementados con *Open5GS* y *srsRAN Project* son capaces de transmitir a tasas más que aceptables cuando se utiliza un dispositivo adecuado como UE.

10 F. Pedreira, B. Tió, L. Barboni and C. Rattaro

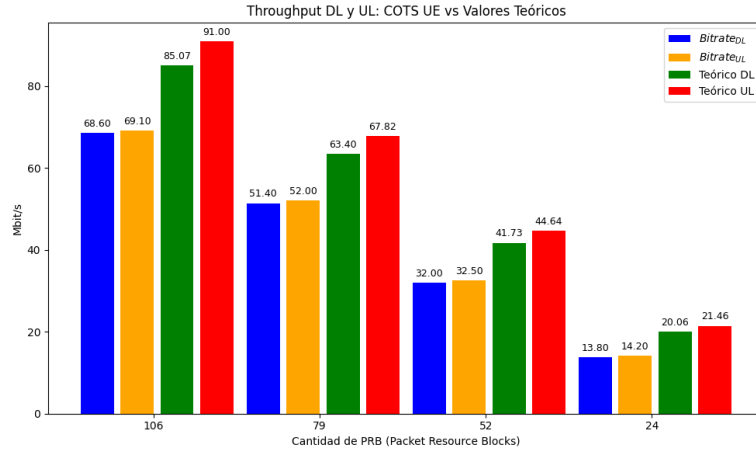


Fig. 5: Throughputs alcanzados por COTS UE comparados con valores teóricos, agrupados por cantidad de PRB

4.2 Uso de recursos hardware

Con el fin de estudiar el comportamiento a nivel de hardware, profundizando en otras posibles causas del bajo desempeño inicial, se evaluó el rendimiento de los procesos correspondientes a la ejecución del core Open5GS en un contenedor Docker y la gNB ejecutando el módulo de *srsRAN*, durante las pruebas detalladas en la sección 3.2. De esta manera se midieron los recursos requeridos por cada uno en las diferentes configuraciones utilizadas. La referencia usada fue el total de recursos de hardware de las PCs, detallados en la sección 3.1.

Por un lado, el núcleo de red Open5GS mantuvo un uso de recursos constante, con un tope de 1.5% de uso de CPU, y un máximo de uso de memoria RAM de 500 MB. Dados tanto la disponibilidad de CPU como el máximo de 16 GB de RAM, esto significa que la ejecución del mismo es sumamente liviana para la PC. A su vez, se observó que estos valores se mantenían estables en las diferentes pruebas realizadas, lo cual era esperado. Esto se debe a que las diferentes configuraciones usadas en la gNB no modificaban parámetros que impactaran en el núcleo de red.

En cuanto a la ejecución de la gNB, el uso de hardware si varió en las pruebas realizadas, en particular al variar la frecuencia de muestreo mediante el uso de diferentes anchos de banda. Esto se debe a la necesidad de procesar una mayor cantidad de paquetes por segundo para mantener esta frecuencia de muestreo transmitida en el SDR. De lo contrario suceden casos de *underflow*, en que la interfaz del SDR consume paquetes más rápido de lo que el proceso de la gNB los puede entregar, llevando a errores y retrasos en los que el SDR debe transmitir paquetes vacíos.

Por lo tanto, el recurso determinante en este caso fue el acceso a la CPU. Este aumentó al aumentar la frecuencia de muestreo, mientras que el uso de memoria RAM se mantuvo estable en 2.2 GB. Otros factores asociados a la memoria del proceso como la memoria de *swap*, se mantuvieron también estables, por lo que se considera que esta no fue un factor determinante. El uso de CPU observado fluctuaba durante la ejecución, pero se observaron en promedio los valores presentes en la tabla 2.

Table 2: Porcentaje de uso de CPU aproximado.

BW	Uso de CPU (%)
5	60
10	80
15	90
20	110

De igual manera, se observó que el uso de CPU dado es aceptable no solo en las PCs utilizadas, si no que debería también ser posible de ejecutar en equipos de menor rendimiento ajustando los parámetros de configuración necesarios. Los demás factores como uso de memoria RAM son lo suficientemente bajos como para que estos puedan ser ejecutados en equipos con menos recursos.

5 Conclusiones

Los resultados obtenidos reflejan que la implementación *end-to-end* de *OpenAir-Interface* permite alcanzar mayores velocidades de *throughput* en comparación con su análoga basada en *srsRAN Project*. No obstante, el rendimiento obtenido con esta es suficiente para realizar pruebas utilizando diferentes equipos y configuraciones. Además, destaca por su facilidad de uso y configuración, lo que lo convierte en una opción atractiva para usuarios que no necesariamente estén familiarizados con la disciplina, facilitando también su implementación en otras áreas.

Asimismo, se observó que los valores de *bitrate* alcanzados al utilizar un dispositivo comercial como equipo de usuario superan ampliamente los logrados con *srsUE 4G*, logrando porcentajes elevados del valor teórico esperado. Esto sugiere que la principal limitante radica en la implementación del UE, evidenciando que tanto el núcleo como el desarrollo de la radiobase pueden ofrecer un buen desempeño si se utiliza un equipo de usuario adecuado. Queda planteado como trabajo futuro un análisis detallado de los protocolos o servicios que *srsUE 4G* no soporta y que podrían estar limitando las tasas de transferencia. Es importante señalar que el *roadmap* de *srsRAN Project* no contempla actualmente el desarrollo de un *srsUE 5G*.

12 F. Pedreira, B. Tió, L. Barboni and C. Rattaro

Por otro lado, se observó que *srsRAN Project* incluye herramientas oficiales que permiten su uso en configuraciones alternativas, como los *splits* funcionales [4]. Esto abre nuevas posibilidades para estudios más detallados y enfoques alternativos, tales como el desarrollo de versiones portables de la maqueta utilizando hardware de menor costo.

Actualmente, el grupo de trabajo se encuentra profundizando en este aspecto, evaluando los requerimientos mínimos necesarios para esta solución e implementando *splits* del tipo CU/DU con dicho propósito. Se busca desarrollar una versión portable de la gNB que sea fácilmente trasladable y que aloje parte de su procesamiento en la nube.

References

1. 3rd Generation Partnership Project (3GPP): Nr; nr and ng-ran overall description; stage-2. Tech. rep. (2018)
2. Dang, S., Amin, O., Shihada, B., Alouini, M.S.: On challenges of sixth-generation (6g) wireless networks: A comprehensive survey of requirements, applications, and security issues. *IEEE Communications Surveys & Tutorials* **23**(1), 6–36 (2021)
3. Ettus research, usrp x300 and x310 x series reference sheet, <https://www.ettus.com/all-products/x310-kit/>
4. L. M. P. Larsen, A.C., Christianse, H.L.: A survey of the functional splits proposed for 5g mobile crosshaul networks. *IEEE Communications Surveys & Tutorials* **21**(1), 146 (2019)
5. Mitola, J.I.: *Software Radio Architecture: Object-Oriented Approaches to Wireless Systems Engineering*. Wiley-Interscience (2000)
6. OpenAirInterface Software Alliance: Openairinterface: 5g software alliance for democratising wireless innovation. <https://www.openairinterface.org> (2024), accessed: 2024-12-15
7. P. Vázquez, W. Peña, W.P.L.I., Rattaro, C.: Maq5g: Deployment of a complete 5g standalone network testbed for testing and development. In: 2024 XVI Congreso de Tecnología, Aprendizaje y Enseñanza de la Electrónica (TAEE). pp. 1–7. Malaga, Spain (2024). <https://doi.org/10.1109/TAEE59541.2024.10604930>
8. srsRAN Project: srsran gnb with srsue. *srsRAN Documentation* (2024), <https://docs.srsran.com/projects/project/en/latest/tutorials/source/srsUE/source/-index.html>
9. srsRAN Project: srsran: Open source 4g and 5g radio access network. <https://www.srsran.com> (2024), accessed: 2024-12-15
10. R. P. Alves, J. G. A. da S. Alves, M.R.C.W.O.d.F., Monteiro, V.F.: Experimental comparison of 5G SDR platforms: srsRAN x OpenAirInterface. Master's thesis, Federal University of Ceará (UFC), Fortaleza, Ceará, Brazil (2024)