ScuDo
Scuola di Dottorato ~ Doctoral School
WHAT YOU ARE, TAKES YOU FAR
XXXVIIth Cycle

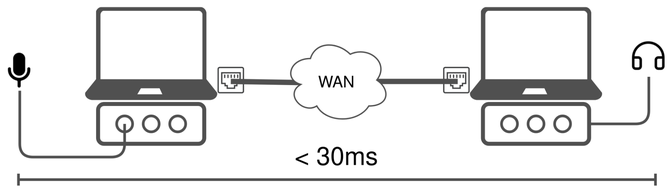
Improving user experience in networked music performance

Matteo Sacchetto

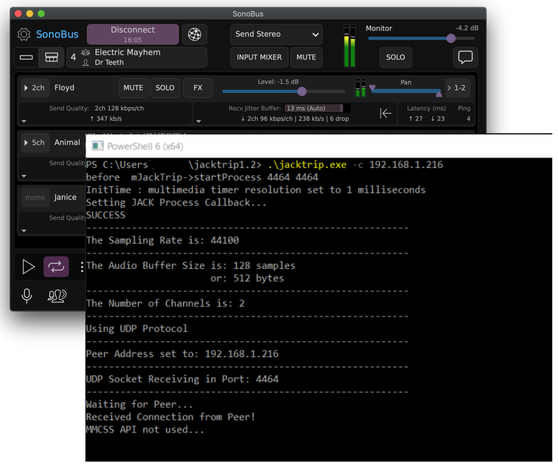
Supervisors: Prof. Cristina Rottondi, Prof. Andrea Bianco

Research context and motivation

- NMP is "a **real-time** interaction **over the network** which enables musicians to play together **as if they were in the same room**"
- In NMP latency and **audio quality** are the two most important factors



  - Latency < 30ms
  - **Uncompressed audio** (20Hz-20kHz).
- Most of the NMP software available today has **considerable complexity of use**, which pushes users to fall back on traditional videoconferencing applications
- Standard videoconferencing platforms lack in performance
  - Latency > 100ms
  - **Voice optimized codecs/PLC**
  - Heavy processing (noise suppression, echo cancellation, …)
- Traditional NMP solutions **lack in usability and accessibility**
  - Difficult installation
  - **Non-trivial connection setup**
  - **Complex/not accessible interface**



Addressed research questions/problems

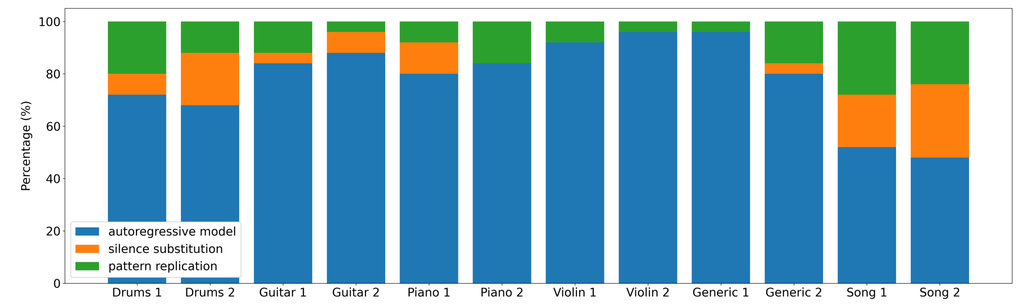
- Traditional packet loss concealment (PLC) leads to **suboptimal audio quality**
  - Silence substitution

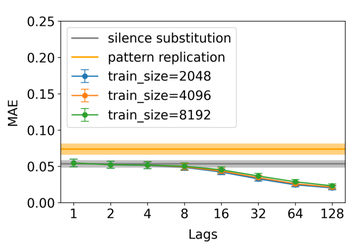
0.9	0.8	...	0.8	0.5	0	0	...	0	0

  - Pattern replication

0.9	0.8	...	0.8	0.5	0.9	0.8	...	0.8	0.5

- AR models proved to improve the perceived audio quality

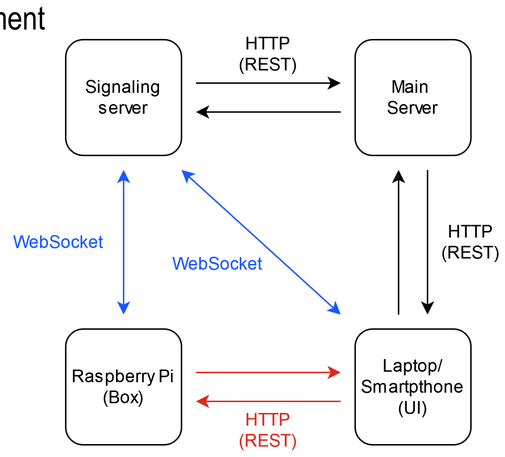




But can they run in real-time on a Raspberry Pi 4B?

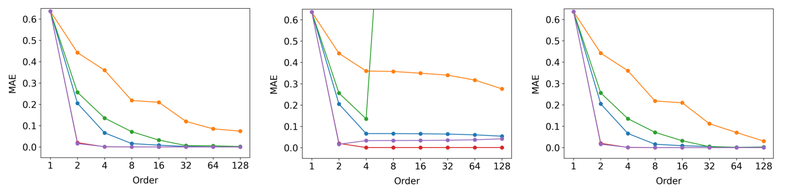
- In the first version of the NMP solution under development the server-side architecture and the UI were designed with a **focus on a music school context**
  - This context was too limited to support all the most common use-cases
  - The architecture had some pitfalls
    - **Interaction between UI and Box**
    - Connection handling

Addressing those limitations requires an architectural change



Novel contributions

- **AR models implemented in C++ using Burg's method**
  - **3 variants** (standard, optimized denominator, hybrid)
  - **2 different data types** (double, long double)
  - Experimented with compilation flags and **error-free floating-point transformations**



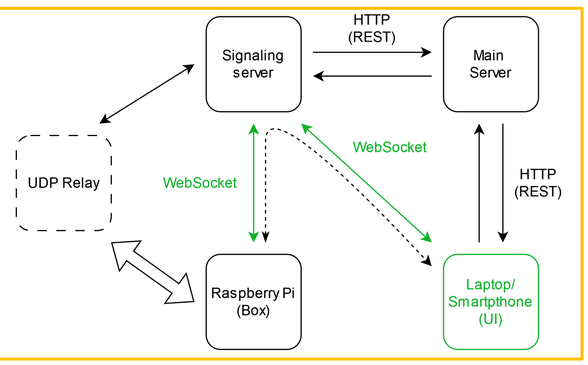
  - **MAE of the prediction of 2048 samples** of a 2kHz sinewave

	Fit time (ms)					Prediction time (ms)	
Algorithm	n=512	n=1024	n=2048	n=4096	n=8192	ρ=64	ρ=128
Burg's method	0.178	0.362	0.746	1.520	3.024	0.028	0.058
Denominator optimization	0.091	0.182	0.378	0.771	1.528		
Hybrid denominator	0.102	0.205	0.424	0.866	1.720		
Burg's method (`-ffast-math`)	0.114	0.228	0.477	1.007	1.985		
Den. opt. (`-ffast-math`)	0.069	0.136	0.292	0.624	1.227	0.022	0.050
Hybrid den. (`-ffast-math`)	0.075	0.148	0.315	0.672	1.320		
Burg's method (compensated)	0.795	1.628	3.327	6.706	13.385		
Den. opt. (compensated)	0.473	0.962	1.967	3.945	7.878	0.060	0.124
Hybrid den. (compensated)	0.515	1.049	2.139	4.292	8.567		

  - Window size of 128 samples at 44.1kHz (= **2.9ms**)

AR Models can run in real-time on a Raspberry Pi 4B

- Server-side architecture was **redesigned considering a wider variety of use cases**
  - Connecting between UI and boxes **is relayed through the servers**
  - Connection between boxes now could support more complex scenarios



Adopted methodologies

- **Autoregressive Models:** type of process where the forecast variable is expressed as a linear combination of its past values
  - $AR(\rho) = y_t = c + a_1 y_{t-1} + a_2 y_{t-2} + \dots + a_\rho y_{t-\rho} + \varepsilon_t$
- **Burg's method**: a method to compute the reflection coefficients (k_i) without needing to compute the autocorrelation matrix. These are then used in the Levinson recursion.
- **Burg's denominator optimization**: an optimization of the Burg's method which can compute the denominator of the reflection coefficient formula in a recursive way.
- **Levinson recursion**: a method to compute the AR model coefficients from the k_i
  - $[a_{i,0}, a_{i,1}, \dots, a_{i,i}] = [a_{i-1,0}, a_{i-1,1}, \dots, a_{i-1,i-1,0}] + k_i[0, a_{i-1,i-1}, a_{i-1,i-2}, \dots, a_{i-1,0}]$
- **Error-free floating-point transformations**: transformations to avoid error propagation in floating- point operations
  - TwoSum, TwoProductFMA and Dot2
- **Microservices design principles**: develop the server architecture in compliance with the microservices design guidelines

Future work

- Find a **deployment strategy** for AR models
- **Deploy AR models in a real scenario** to assess their performance
- **Finish the development of the server architecture and integrate it with the box**
- Add **NAT traversal** capabilities
- Add additional capabilities, such as **remote mixing** or video streaming

Publications

- Published works: 1 journal, 2 conferences
- Submitted works: 1 journal, 3 conferences
- "Web-Based Networked Music Performances via WebRTC: a Low-latency PCM Audio Solution", JAES, https://dx.doi.org/10.17743/jaes.2022.0021
- "Using Autoregressive Models for Real-Time Packet Loss Concealment in Networked Music Performance Applications", AM22 , https://dx.doi.org/10.1145/3561212.3561226

Politecnico di Torino 1859

PhD program in
Electrical, Electronics and Communications Engineering