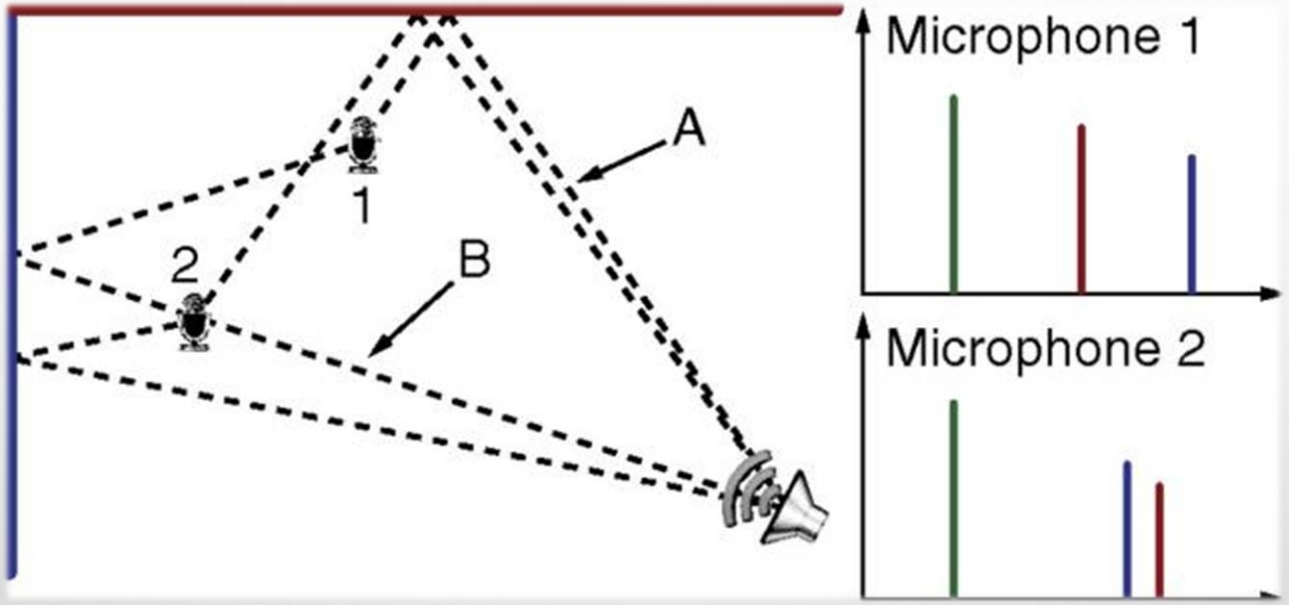
QR UPDATE ON BEAMFORMING ALGORITHM

AUTHOR: FRANCISCO JAVIER ALVENTOSA RUEDA

DIRECTED BY: PEDRO ALONSO JORDÁ AND ANTONIO MANUEL VIDAL MACIÁ

BEAMFORMING: SIGNAL MODEL



- THE SIGNAL CAPTURED BY THE MICROPHONE IS THE SIGNAL EMITTED BY THE LOUDSPEAKER *S(N)*, FILTERED BY THE ROOM IMPULSE RESPONSE *H(N)*.

$$X(N) = H(N)*S(N)+NOISE$$

- FOR MORE LOUDSPEAKERS AND MICROPHONES, THE SYSTEM IS MIMO (MULTIPLE INPUT MULTIPLE OUTPUT).

BEAMFORMING: HARDWARE (NVIDIA JETSON TK1)

FEATURES:

- TEGRA K1 SOC:
  - KEPLER GPU WITH 192 CORES.
  - 4+1 QUAD-CORE ARM CORTEX-A15 CPU.
- 2GB RAM.
- 16GB EMMC MEMORY.
- 1 MINI PCI.
- 1 USB 3.0.
- 1 SATA.
- 1 HDMI.

BEAMFORMING: ALGORITHM ANALYSES

- GOAL: TO WORK IN REAL TIME.
- OVER 70% OF ALGORITHM COST IS QR FACTORIZATION OF MATRIX X.
  - WE FOCUSED OWN EFFORT TO REDUCE THIS COST.
- WE EMPLOYED DIFFERENT HPC LIBRARIES LIKE OPENBLAS, PLASMA, CUBLAS AND MAGMA.
- BEAMFORMING ALGORITHM IS A ITERATIVE ALGORITHM, FOR THIS REASON, WE NEED TO CALCULATE QR FACTORIZATION ALL THE TIME.

QR UPDATE

- QR UPDATE ALLOWS TO RECALCULATE THE NEWS Q AND R FACTORS FROM EXISTING ONES, THAT IS, IT AVOID HAVING TO CALCULATE THEM COMPLETELY.

- THE PROCEDURE FOLLOWED TO UPDATE THE QR DECOMPOSITION HAS BEEN EXTRACTED FROM [1].

- THIS ALGORITHM HAS BEEN DIVIDED IN TWO PARTS: **DELETE ROWS** AND **ADD ROWS**.

[1] Sven Hammarling and Craig Lucas, "Updating the QR factorization and the least squares problem" in MIMS EPrint, 2008.111.

QR UPDATE – DELETE ROWS

$$X = (QH)(H^T R) = \begin{pmatrix} I_k & 0 \\ 0 & Q' \end{pmatrix} \begin{pmatrix} V \\ R' \end{pmatrix} . \quad (1)$$

- WE HAVE TWO MATRICES Q & R THAT REPRESENT THE QR DECOMPOSITION OF MATRIX X.
- MATRIX H HAVE A K HOUSEHOLDER VECTORS TO DO 0 ON THE FIRST K ROWS OF MATRIX Q.
- I_k IS A IDENTITY MATRIX.
- Q' AND R' REPRESENT THE QR DECOMPOSITION OF MATRIX X', THAT IS, MATRIX X WITHOUT THEIR K FIRST ROWS.
- MATRIX $V \in \mathbb{R}^{k \times n}$ CONTAINS THE FIRST K ROWS OF MATRIX X.

QR UPDATE – ADD ROWS

$$X_1 = \left(\begin{pmatrix} Q' & 0 \\ 0 & I_k \end{pmatrix} G \right) \left(G^T \begin{pmatrix} R' \\ W \end{pmatrix} \right) = Q_1 R_1 ,$$

- X1 IS MATRIX X' WITH K NEW ROWS AT THE END (W MATRIX).
- G MATRIX CONTAIN THE GIVENS TRANSFORMATION TO OBTAIN THE QR FACTORIZATION OF MATRIX X1.
- I_k IS A IDENTITY MATRIX.

RESULTS

Table 1: Time for the QR decomposition of matrix X (2240 × 960) using 1 to 4 cores of the NVIDIA Jetson TK1 with OpenBLAS.

N. cores	Exec. time
1	1.12 s.
2	0.63 s.
3	0.52 s.
4	0.39 s.

- HIGH COST OF QR UPDATE VS NEW QR.
- A FEW ROWS CAN BE ACTUALIZED.
- QR UPDATE NOT HAVE MORE PARALLELISM.

Table 2: Time for the QR updating of matrix X (2240 × 960) using k rows in the NVIDIA Jetson TK1.

N. rows (k)	5	10	15	20	25
Exec. time	0.26 s.	0.47 s.	0.69 s.	0.90 s.	1.12 s.
N. rows (k)	30	35	40	45	50
Exec. Time	1.33 s.	1.55 s.	1.77 s.	1.98 s.	2.20 s.

CONCLUSIONS AND FUTURE WORKS

- OVER 70% COST BEAMFORMING ALGORITHM IS QR FACTORIZATION.
- QR UPDATE HAVE A HIGH COMPUTATIONAL COST.
- EMPLOY DIFFERENT STRATEGIES FOR OPTIMIZE THE QR UPDATE (QR MULTICORE STRATEGY).