



Parallelization of a Denoising Algorithm for Tonal Bioacoustic Signals using OpenACC Directives

J. Castro, E. Meneses

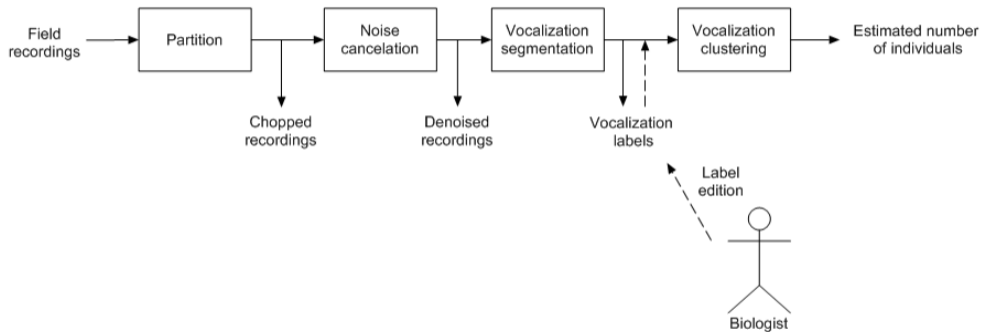
Advanced Computing Laboratory
Costa Rica National High Technology Center

July, 2018

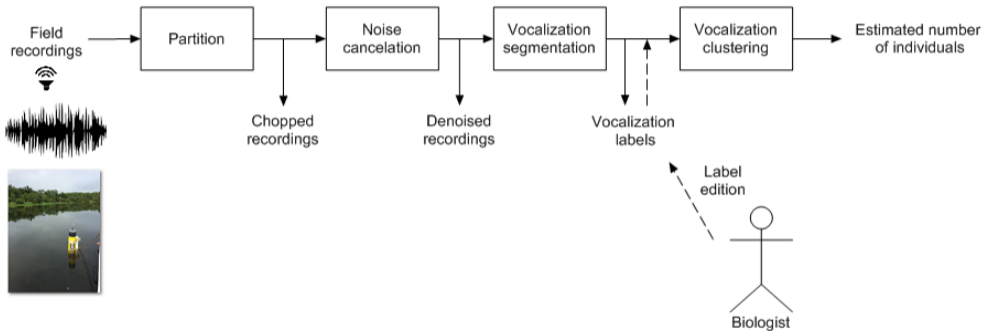
Motivation



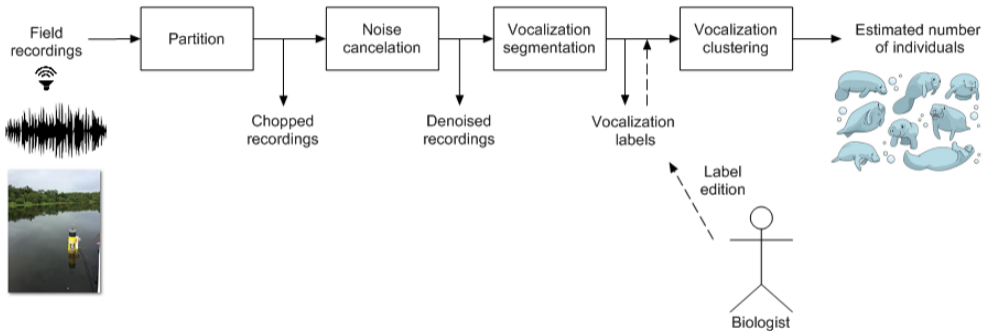
Automatic Manatee Count Method



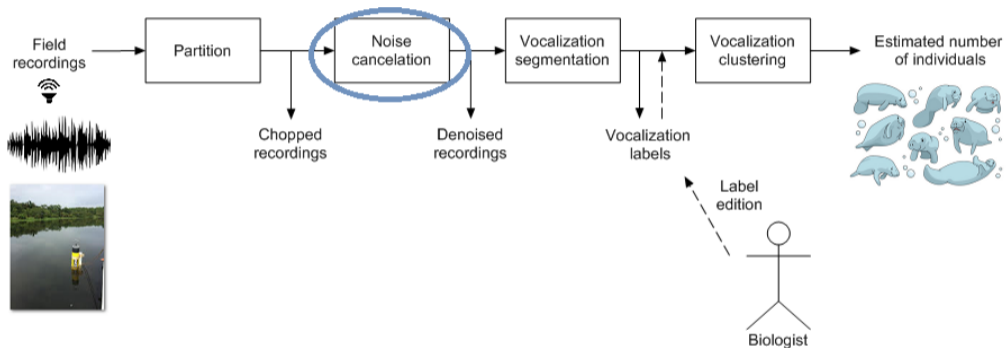
Automatic Manatee Count Method



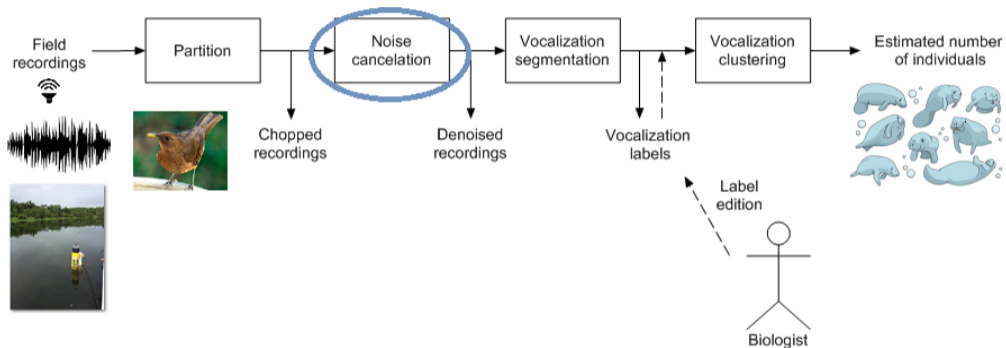
Automatic Manatee Count Method



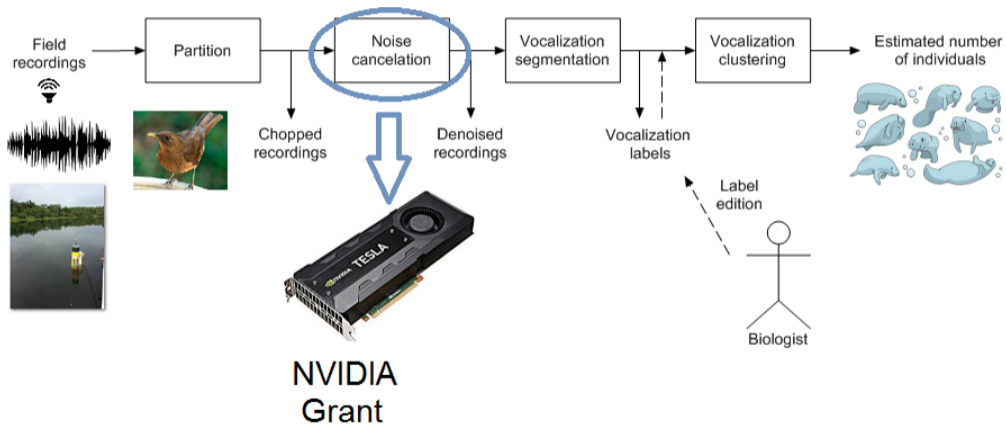
Automatic Manatee Count Method



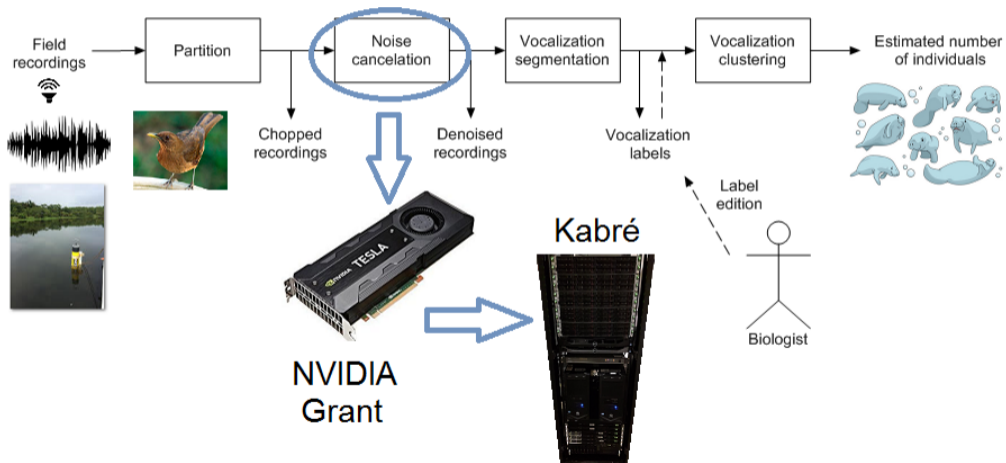
Automatic Manatee Count Method



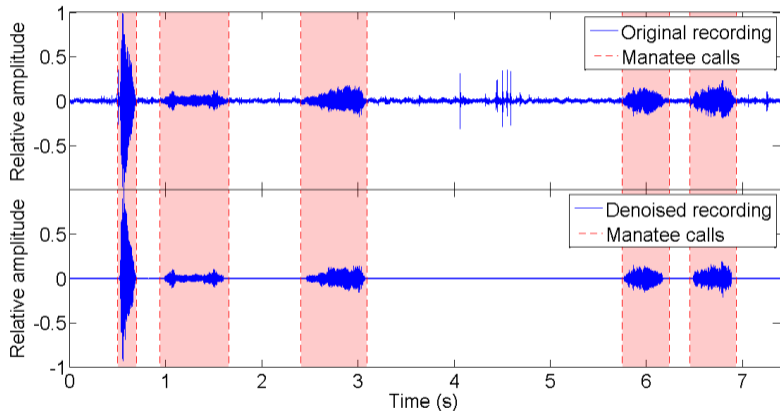
Automatic Manatee Count Method



Automatic Manatee Count Method



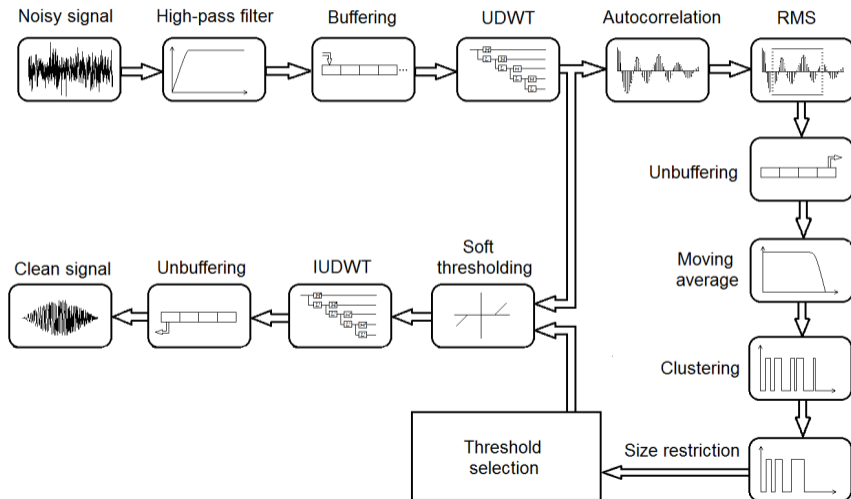
Denosing method



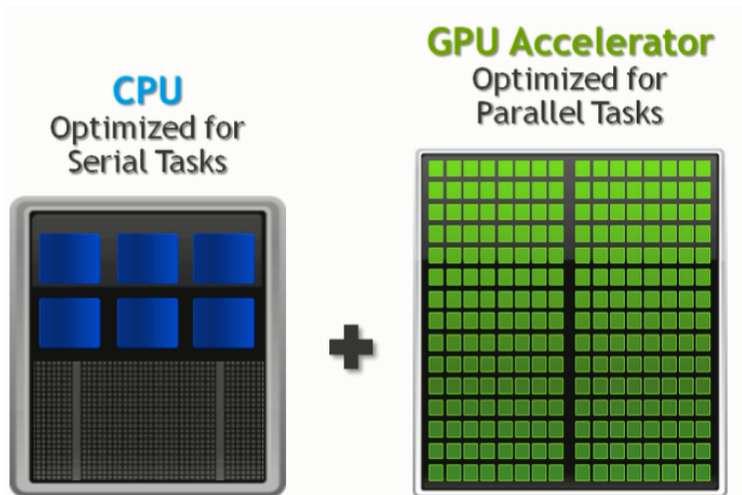
● Original 🔊

● Denoised 🔊

Denoising method



Why GPUs?



Why OpenACC?

OpenACC

Directives for Accelerators



OpenCL



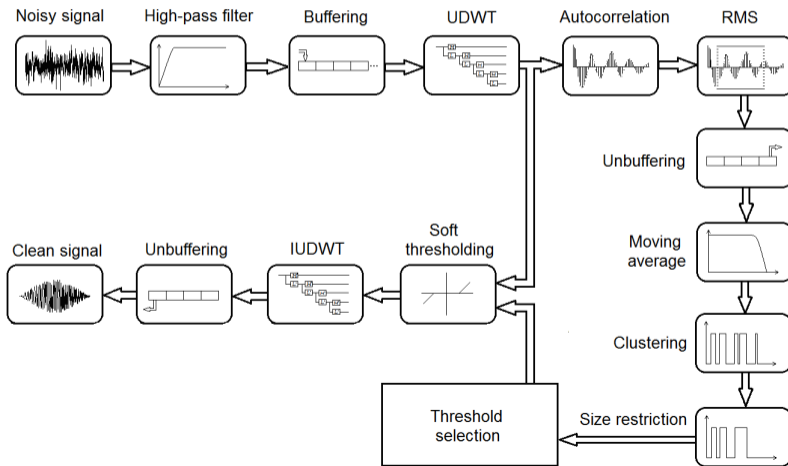
nVIDIA®

CUDA®

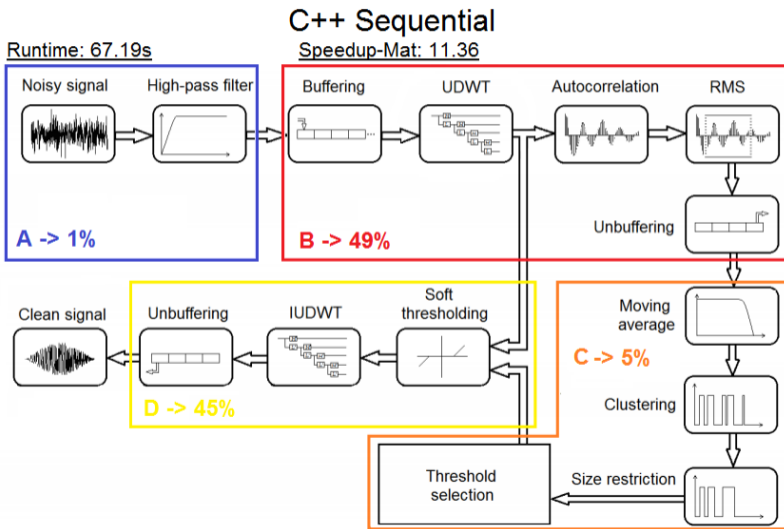
Original implementation

Matlab Sequential

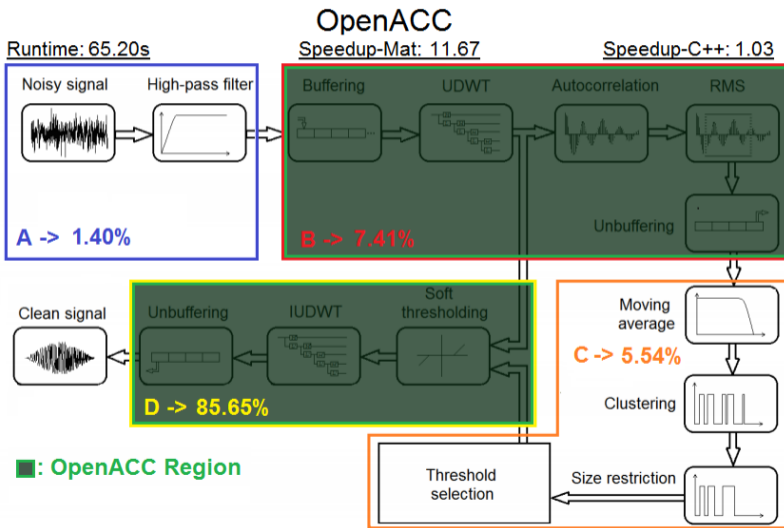
Runtime: 761s



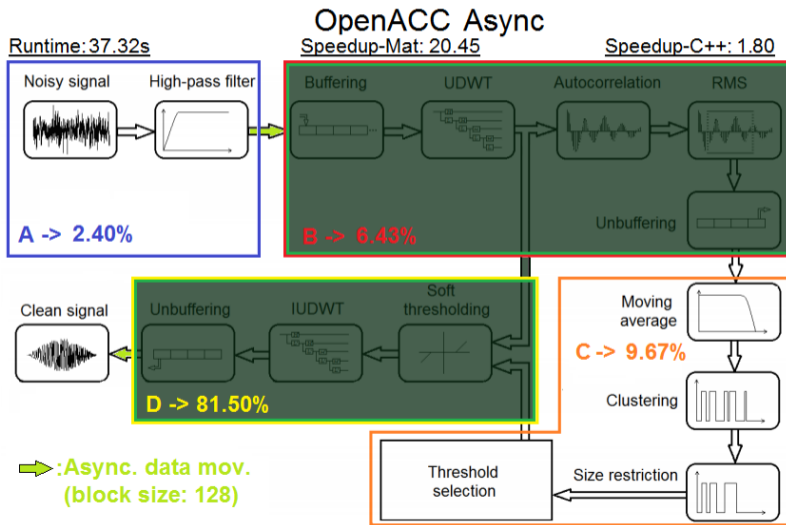
Profiling in C++



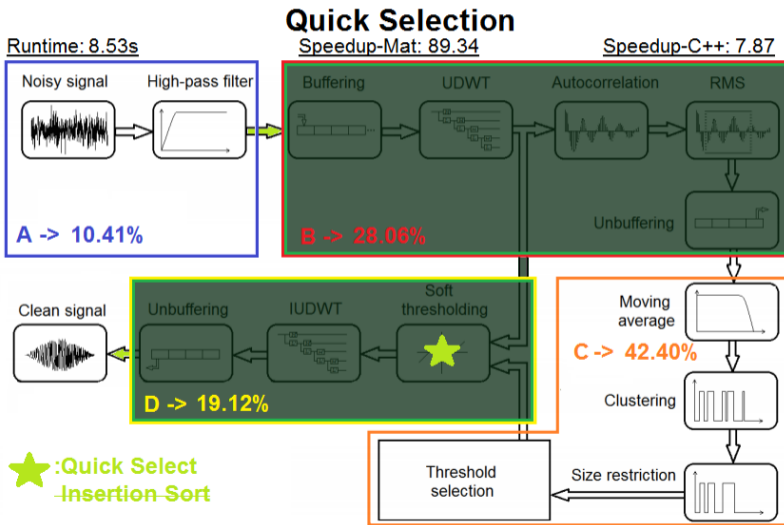
Parallelizing with OpenACC



Asynchronous data movement



Quick Select Algorithm



Dynamic clustering parallelization

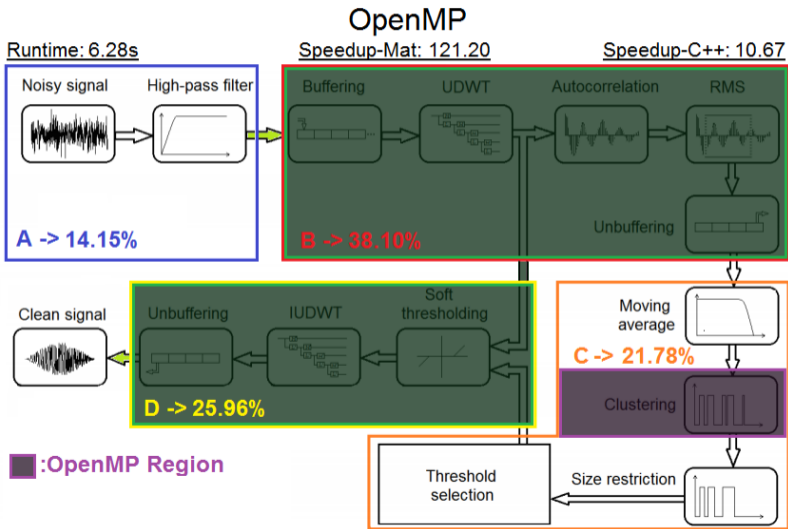
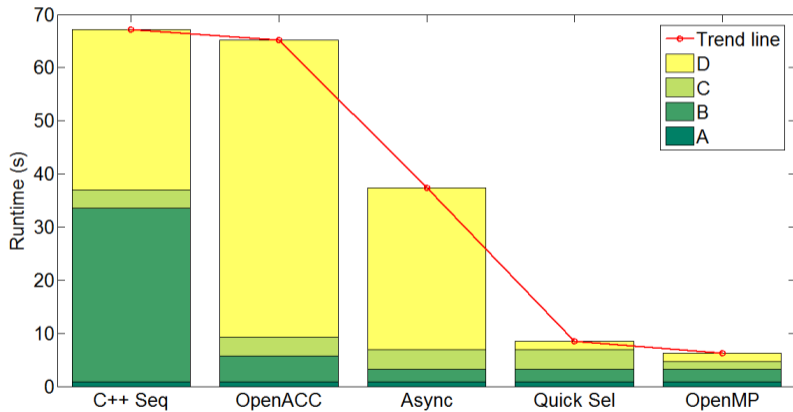


Table: Ten runs using a nine-minute manatee vocalization recording.

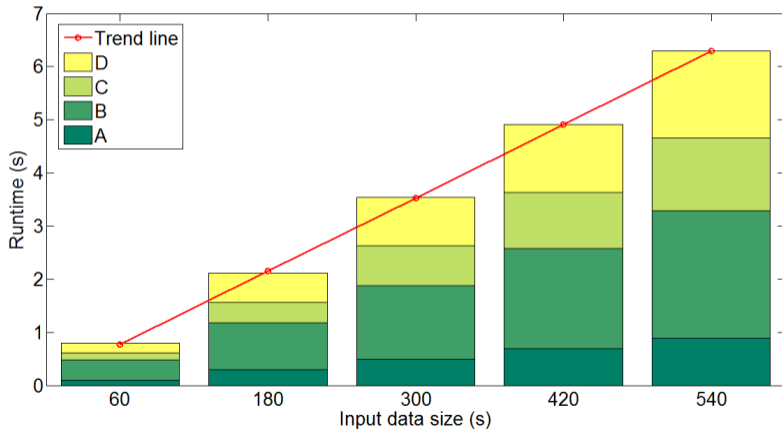
Code	Arch	Duration (s)	Speed-up _M	Speed-up _C
Matlab	CPU	761.714 ± 1.73	1	0.09
C++ Seq	CPU	67.07 ± 0.11	11.36	1
OpenACC	GPU	65.29 ± 0.07	11.67	1.03
Async	GPU	37.24 ± 0.05	20.45	1.80
Quick Sel	GPU	8.53 ± 0.01	89.34	7.87
OpenMP	GPU	6.28 ± 0.01	121.20	10.67
Multicore	CPU	14.51 ± 0.02	52.49	4.62

- CPU: Intel Xeon CPU E3-1225 v5 @ 3.30GHz (quadcore)
- GPU: Tesla k40c

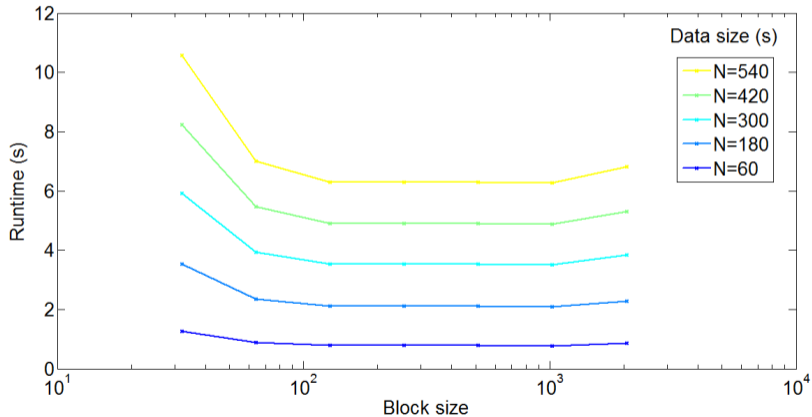
Runtime decomposition



Scalability: input data



Scalability: block size



Speed-up: C++ optimized algorithm

Table: Ten runs using a nine-minute manatee vocalization recording.

Code	Arch	Duration (s)	Speed-up
C++ Seq QS	CPU	46.13 ± 0.11	1
OpenACC QS	GPU	15.32 ± 0.06	3.01
Async QS	GPU	8.53 ± 0.01	5.41
OpenMP QS	GPU	6.28 ± 0.01	7.34
Multicore QS	CPU	14.51 ± 0.02	3.18

- CPU: Intel Xeon CPU E3-1225 v5 @ 3.30GHz (quadcore)
- GPU: Tesla k40c

¡Thank you!

- First parallelization of a bioacoustic algorithm using OpenACC
- The algorithm scales linearly
- Maximum speed-up $S_M = 121.2$ and $S_C = 10.67$



¿Questions?

<http://www.cenat.ac.cr/cnca>

jcastro@cenat.ac.cr