

# Package ‘RbowtieCuda’

September 30, 2025

**Type** Package

**Title** An R Wrapper for nvBowtie and nvBWT, a rewritten version of Bowtie2 for cuda

**Version** 1.1.9

**Date** 2025-09-25

**Description** This package provides an R wrapper for the popular Bowtie2 sequencing read aligner, optimized to run on NVIDIA graphics cards. It includes wrapper functions that enable both genome indexing and alignment to the generated indexes, ensuring high performance and ease of use within the R environment.

**License** BSD\_3\_clause + file LICENSE

**Depends** R (>= 4.5.0)

**Encoding** UTF-8

**Imports** methods

**Suggests** knitr, rmarkdown, RUnit, BiocGenerics

**SystemRequirements** C++20, GNU make, cmake, CUDA Toolkit(>=10), MSVC, libthrust-dev, libcub-dev, gcc, g++

**Archs** x64

**RoxygenNote** 7.3.2

**biocViews** Sequencing, Alignment, Preprocessing, Coverage

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**NeedsCompilation** yes

**Author** c(Jacopo Pantaleoni [aut],  
Nuno Subtil [aut],  
Samuel Simon Sanchez [aut],  
Franck RICHARD [aut, cre],  
role = c("`aut", "`cre"))

**Maintainer** Franck RICHARD <franck.richard@winstars.net>

**BugReports** <https://github.com/FranckRICHARD01/RbowtieCuda/issues>

**URL** <https://github.com/FranckRICHARD01/RbowtieCuda>,  
<https://belacqua-labo.ovh/bioinformatic/RbowtieCuda>

**git\_url** <https://git.bioconductor.org/packages/RbowtieCuda>

**git\_branch** devel  
**git\_last\_commit** e64d18a  
**git\_last\_commit\_date** 2025-09-12  
**Repository** Bioconductor 3.22  
**Date/Publication** 2025-09-30

**Contents**

|                            |          |
|----------------------------|----------|
| .callbinary2 . . . . .     | 2        |
| nvbio_tests . . . . .      | 3        |
| nvBowtie . . . . .         | 3        |
| nvBowtie_usage . . . . .   | 4        |
| nvBowtie_version . . . . . | 5        |
| nvBWT . . . . .            | 5        |
| <b>Index</b>               | <b>7</b> |

---

|              |                                      |
|--------------|--------------------------------------|
| .callbinary2 | <i>Make system call for binaries</i> |
|--------------|--------------------------------------|

---

**Description**

Function that makes a system call for the RbowtieCuda binaries. Note that it is not designed to be used outside the package.

**Usage**

.callbinary2(bin1, args1, path = NULL)

**Arguments**

- bin1                      Character. The binary used for the system call.
- args1                    Character. The arguments to pass to the binary.
- path                     Character Optional: If passed to function, returns the path.

**Value**

An invisible Integer of call status. The value is 0 when there is not any mistakes.

**Author(s)**

Franck RICHARD

**Examples**

.callbinary2(bin1 = "nvBowtie", args1 = "--version")

---

`nvbio_tests`*Print unit tests*

---

**Description**

Calling `nvbio_tests()` performs alignment tests

**Usage**

```
nvbio_tests()
```

**Value**

The return value is the string "nvbio test passed." when there is not any mistakes.

**Author(s)**

Franck RICHARD

**Examples**

```
nvbio_tests()
```

---

`nvBowtie`*nvBowtie align function*

---

**Description**

`nvBowtie` is a GPU-accelerated re-engineering of Bowtie2, a very widely used short-read aligner.

**Usage**

```
nvBowtie(index, output_file, options, seq1, seq2 = NULL)
```

**Arguments**

|                          |                                                                                                                                                                                          |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>index</code>       | Character vector. Index file created by nvBWT.                                                                                                                                           |
| <code>output_file</code> | Character vector. The alignment output file (.sam o .bam)                                                                                                                                |
| <code>options</code>     | Character vector. Specify additional options here to customize the processing. For a complete list of available options, type <code>nvBowtie_usage()</code> .                            |
| <code>seq1</code>        | Character vector. For single-end sequencing, it contains sequence file paths. For paired-end sequencing, it can be file path with #1 mates paired with file paths in <code>seq2</code> . |
| <code>seq2</code>        | Character vector. It contains file paths with #2 mates paired with file path in <code>seq1</code> . For single-end sequencing files, it must be NULL.                                    |

## Details

We've introduced several new features to nvBowtie. You can now perform alignments using the WFA method by including the `--wfa` (or `--scoring wfa`) parameter. The WFA method requires a large amount of RAM on the graphics card. We therefore recommend using an Nvidia card with 8GB or more. Please note that this feature is still experimental; it currently supports only end-to-end alignments and does not yet allow customization of scoring parameters. By default, it uses the following scoring: `match:0, mismatch:1, gap_open:1` and `gap_ext:1`. Additionally, the `--cache-writes` parameter optimizes disk write operations, resulting in faster alignments. This functionality requires 4GB of RAM and is limited to paired-end alignments.

## Value

An invisible Integer of call status. The value is 0 when there is not any mistakes Otherwise the value is non-zero.

## References

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature methods*, 9(4), 357-359.

## Examples

```
td <- tempdir()

## Building index
fa_file <- system.file(package='RbowtieCuda', 'extdata', 'bt2', 'refs', 'lambda_virus.fa')
nvBWT(myinput=fa_file, output=file.path(td, 'index'), options='')

## Alignments
read_1 <- system.file(package='RbowtieCuda', 'extdata', 'bt2', 'reads', 'reads_1.fastq')
read_2 <- system.file(package='RbowtieCuda', 'extdata', 'bt2', 'reads', 'reads_2.fastq')

## Sam file created with paired-end files (paired mates are here reverse-forward)
nvBowtie(file.path(td, 'index'), file.path(td, 'my_result.sam'), options='--rf', seq1=read_1, seq2=read_2)

## Bam file created with single-end file
nvBowtie(file.path(td, 'index'), file.path(td, 'my_result.bam'), options='', seq1=read_1)
```

---

nvBowtie\_usage

*Print available arguments that can be passed to nvBowtie()*

---

## Description

Calling `nvBowtie_usage()` prints available arguments that can be passed to the `nvBowtie()` function.

## Usage

```
nvBowtie_usage()
```

## Value

An invisible Integer of call status. The value is 0 when there is not any mistakes Otherwise the value is non-zero.

**Author(s)**

Franck RICHARD

**References**

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. Nature methods, 9(4), 357-359.

**Examples**

```
nvBowtie_usage()
```

---

|                  |                                  |
|------------------|----------------------------------|
| nvBowtie_version | <i>Print version information</i> |
|------------------|----------------------------------|

---

**Description**

Calling nvBowtie\_version() displays the version information of the RbowtieCuda package in use.

**Usage**

```
nvBowtie_version()
```

**Value**

An invisible Integer of call status. The value is 0 when there is not any mistakes.

**Author(s)**

Franck RICHARD

**Examples**

```
nvBowtie_version()
```

---

|       |                                                                    |
|-------|--------------------------------------------------------------------|
| nvBWT | <i>nvBWT : builds the BWT indices of the reference FASTA files</i> |
|-------|--------------------------------------------------------------------|

---

**Description**

This function can be use to call wrapped nvBWT binary.

**Usage**

```
nvBWT(myinput, output, options = NULL)
```

**Arguments**

|         |                                                                         |
|---------|-------------------------------------------------------------------------|
| myinput | Character vector. A path to a .fa file                                  |
| output  | Character vector. A path to a index file used for the alignment output. |
| options | Character vector. Options Example: --verbosity                          |

## Details

Available options:

|    |  |                |               |                                     |
|----|--|----------------|---------------|-------------------------------------|
| -v |  | --verbosity    | int (0-6) [5] | // select the verbosity level       |
| -m |  | --max-length   | int [inf]     | // clamp input length               |
| -b |  | --byte-packing | [default]     | // output a byte-encoded .pac file  |
| -w |  | --word-packing |               | // output a word-encoded .wpac file |
|    |  |                |               | // (more efficient)                 |
| -c |  | --crc          |               | // compute CRCs                     |
| -d |  | --device       |               | // select a cuda device             |

## Value

An invisible Integer of call status. The value is 0 when there is not any mistakes Otherwise the value is non-zero.

## Author(s)

Franck RICHARD

## References

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature methods*, 9(4), 357-359.

## Examples

```
td <- tempdir()
fa_file <- system.file(package='RbowtieCuda', 'extdata', 'bt2', 'refs', 'lambda_virus.fa')
nvBWT(myinput=fa_file, output=file.path(td, 'index'), options='')
```

# Index

`.callbinary2`, [2](#)

`nvbio_tests`, [3](#)

`nvBowtie`, [3](#)

`nvBowtie_usage`, [4](#)

`nvBowtie_version`, [5](#)

`nvBWT`, [5](#)