



**Ricardo Alexandre do Rosário Ribeiro**

Master of Science

## **Protein docking GPU acceleration**

Dissertation plan submitted in partial fulfillment  
of the requirements for the degree of

Master of Science in  
**Computer Science and Informatics Engineering**

Adviser: Ludwig Krippahl, Full Professor,  
NOVA University of Lisbon

Co-adviser: Hervé Paulino, Associate  
Professor, NOVA University of Lisbon



FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE NOVA DE LISBOA

July, 2018



## RESUMO

---

As proteínas são um dos nutrientes indispensáveis à vida na Terra e a um longo prazo no Universo em geral, a capacidade dos seus compostos (que segundo o ramo da biologia/bioquímica se ditam por amino-ácidos) de se juntarem e formarem vastos complexos permite o crescimento físico dos seres-vivos. Contudo, este processo de ajuntamento em termos informáticos apenas pode ser determinado por matrizes tri-dimensionais em situações em que temos vários compostos interligados, já que os mesmos assumem estruturas tri-dimensionais, sendo mais difícil determinar experimentalmente uma previsão correta sobre estes últimos do que numa situação em que temos proteínas singulares pois existem muitas combinações possíveis para a docagem entre os pares, logo é necessário recorrer a ferramentas que oferecem a computação adequada para determinar estas previsões, a fim de poder-se avançar no estudo das interações entre proteínas e na concepção de produtos à base das mesmas.

O presente documento aborda uma proposição para a paralelização do algoritmo BiGGER implementado pelo prof. Ludwig Krippahl e outros professores associados à criação do algoritmo, recorrendo a técnicas de computação acelerada i.e. utilizar o GPU da máquina em que corre o algoritmo para auxiliar o CPU na computação que é necessária para o algoritmo na versão actual, de forma a que com mais recursos à disposição, o tempo de execução do BiGGER baixe drasticamente por consequência do aumento significativo de performance.

Em caso de sucesso, a complexidade futura do algoritmo permitirá a adição de mais vantagens face aos seus concorrentes, e por consequência, uma proposta de valor para quem pretenda utilizar o open-chemera como ferramenta de trabalho eficiente no estudo das interações entre os complexos de proteínas em qualquer máquina que tenha uma placa gráfica com as características adequadas.

**Palavras-chave:** proteínas, docagem de proteínas, computação acelerada, GPU, Bio-Informática, BiGGER

---



## ABSTRACT

---

The dissertation must contain two versions of the abstract, one in the same language as the main text, another in a different language. The package assumes that the two languages under consideration are always Portuguese and English.

The package will sort the abstracts in the appropriate order. This means that the first abstract will be in the same language as the main text, followed by the abstract in the other language, and then followed by the main text. For example, if the dissertation is written in Portuguese, first will come the summary in Portuguese and then in English, followed by the main text in Portuguese. If the dissertation is written in English, first will come the summary in English and then in Portuguese, followed by the main text in English.

The abstract should not exceed one page and should answer the following questions:

- What's the problem?
- Why is it interesting?
- What's the solution?
- What follows from the solution?

**Keywords:** Keywords (in English) ...

---



# ÍNDICE

<b>Lista de Figuras</b>	<b>ix</b>
<b>Lista de Tabelas</b>	<b>xi</b>
<b>Listagens</b>	<b>xiii</b>
<b>Glossário</b>	<b>xv</b>
<b>Siglas</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 A Bit of History . . . . .	1
1.2 Disclaimer . . . . .	1
<b>2 Estado-da-arte</b>	<b>3</b>
2.1 GPU . . . . .	3
2.2 Classificações de algoritmos . . . . .	3
2.3 BiGGER . . . . .	4
2.4 BoGIE . . . . .	5
2.5 Fast Fourier Transform . . . . .	5
2.6 Algoritmos de correlação espacial . . . . .	5
<b>3 A Short L<sup>A</sup>T<sub>E</sub>X Tutorial with Examples</b>	<b>7</b>
3.1 Document Structure . . . . .	7
3.2 Dealing with Bibliogrpahy . . . . .	7
3.3 Inserting Tables . . . . .	7
3.4 Importing Images . . . . .	7
3.5 Floats, Figures and Captions . . . . .	7
3.6 Text Formatting . . . . .	9
3.7 Generating PDFs from L <sup>A</sup> T <sub>E</sub> X . . . . .	9
3.7.1 Generating PDFs with pdflatex . . . . .	9
3.7.2 Dealing with Images . . . . .	10
3.7.3 Creating Source Files Compatible with both latex and pdflatex . .	10
3.8 Equações . . . . .	12





## LISTA DE FIGURAS

2.1	Representação em 2D das matrizes resultantes do segundo passo do BoGIE, as células preenchidas a tracejado diagonal correspondem à matriz de superfície, as com pontos correspondem à matriz <i>core</i> e a núvem com tracejado contínuo representa o corte associado à esfera de van der waals com a proteína localizada ao centro [2] . . . . .	6
3.1	A figure with two sub-figures! . . . . .	8
3.2	Imagem em formato <i>bitmap</i> (JPG) . . . . .	13
3.3	Imagem em formato PDF vectorial . . . . .	14
3.4	Exemplo de utilização de <i>subbottom</i> . . . . .	15



## LISTA DE TABELAS

3.1 Test results summary. . . . .	11
-----------------------------------	----



## LISTAGENS

3.1 Hello World . . . . .	12
---------------------------	----



## GLOSSÁRIO

aliquam	tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris..
computer	An electronic device which is capable of receiving information (data) in a particular form and of performing a sequence of operations in accordance with a predetermined but variable set of procedural instructions (program) to produce a result in the form of information or signals..
cras viverra	metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat..
donec nonummy	pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo..
integer sapien	est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus..
lorem ipsum	dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris..
maecenas lacinia	nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem..
morbi ac	orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus..
morbi dolor	nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum..

nam lacus	libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi..
nam dui	ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo..
name arcu	libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo..
nulla malesuada	porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis..
sed lacinia	nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus..



## SIGLAS

aaa	acornym aaa.
aab	acornym aab.
aba	acornym aba.
abbrev	abbreviation of a longer text.
AEU	adipiscing elit ut.
AFM	aenean faucibus morbi.
AMD	a magna donec.
ANP	ac nunc praesent.
ATG	amet tortor gravida.
AVF	adipiscing vitae felis.
bbb	acornym bbb.
CAS	curabitur auctor semper.
CDG	curabitur dictum gravida.
CEA	congue eu accumsan.
CIV	consectetuer id vulputate.
DIA	duis eget orci.
DNM	dolor nulla malesuada.
DNMC	duis nibh mi congue.
DRN	dignissim rutrum nam.
EII	est iaculis in.
ENE	et netus et.
EPA	eu pulvinar at.
ESQ	eleifend sagittis quis.
ESV	eget sem vel.
ETS	eu tellus sit.

FUP    fringilla ultrices phasellus.

LID    lorem ipsum dolor.

LNE    libero nonummy eget.

LUB    leo ultrices bibendum.

LVU    lectus vestibulum urna.

MAC    mollis ac nulla.

MFA    malesuada fames ac.

MNA    mauris nam arcu.

MTS    morbi tristique senectus.

NDV    nulla donec varius.

NPH    neque pellentesque habitant.

OER    orci eget risus.

PEV    purus elit vestibulum.

PIS    placerat integer sapien.

PQV    pretium quis viverra.

SAO    sit amet orci.

SNE    sem nulla et.

STC    sit amet consectetur.

TEM    turpis egestas mauris.

ULC    ut leo cras.

UPA    ut placerat ac.

VAE    vehicula augue eu.

VMR    viverra metus rhoncus.

xpto    and extension of a xpto xpto xpto xpto xpto xpto xpto xpto xpto xpto  
xpto xpto xpto xpto xpto xpto xpto xpto.





## INTRODUCTION

*This work is licensed under the Creative Commons Attribution-NonCommercial 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/4.0/>.*

### 1.1 A Bit of History

The *novathesis* was originally developed to help MSc and PhD students of the Computer Science and Engineering Department of the Faculty of Sciences and Technology of NOVA University of Lisbon (DI-FCT-NOVA) to write their thesis and dissertations Using  $\text{\LaTeX}$ . These student can easily cope with  $\text{\LaTeX}$  by themselves, and the only need some help in the bootstrap process to make their life easier.

However, as the template spread out among the students from other degrees at FCT-NOVA, the demand for am easier-to-use template as grown. And the template in its current shape aims at answering the expectations of those that, although they are not familiar with programming nor with markup languages, so still feel brave enough to give  $\text{\LaTeX}$  a try and rejoice with the beauty of the texts typeset by this system.

### 1.2 Disclaimer

It is up to you, the student, to read the FCT and/or NOVA regulations on how to format and submit your MSc or PhD dissertation.

This template is endorsed by the FCT-NOVA and even linked from its web pages, but it is not an official template. This template exists to make your life easier, but in the end of the line you are accountable for both the looks and the contents of the document you submit as your dissertation.



## ESTADO-DA-ARTE

## 2.1 GPU

## 2.2 Classificações de algoritmos

Com o passar do tempo cada vez mais algoritmos e respectivas adaptações para simular a docagem dos complexos de proteínas têm surgido, surgindo também formas de os classificar. Um algoritmo pode ser classificado em função da forma que trata a rigidez dos pares de proteínas a juntar ou até mesmo pelo modelo matemático, como por exemplo se aplicam a Fast Fourier Transform ou não. Segundo o relatório de Huang Wenfan sobre o uso de Fast Fourier Transform associado à docagem de proteínas rígidas [3], existem três formas de se classificar um algoritmo de docagem, isto em relação às características das proteínas em si, e não sobre o modelo matemático ou metodologias necessárias para o implementar:

- **Docagem Flexível** Em que ambos os complexos receptor e ligando são considerados como sendo corpos flexíveis e adaptáveis sendo, no entanto, a mesma flexibilidade interpretada pelo algoritmo de forma simplificada ou limitada, e por consequência pode-se aplicar um modelo através de simulações de docagem.
- **Docagem Semi-Flexível** Um dos corpos é considerado rígido e o outro não, em situações normais este tipo de algoritmos trata o ligando como se fosse a proteína flexível, já que esta é mais pequena que a sua parceira, tendo assim uma maior probabilidade de mudar a sua forma, outra justificação tem a ver com os custos de computação serem mais baixos do que se considerarmos os receptores como flexíveis.

- **Docagem Rígida** O par é considerado como sendo flexível na sua integridade, sendo também considerado que na docagem entre os dois corpos uma das proteínas irá acabar por penetrar a outra o que leva a que se tenha de adaptar o conjunto de soluções para o problema em seis dimensões, 3 para a rotação e 3 para a translação.

Estes três conceitos são necessários para que se possa entender melhor os algoritmos nas secções seguintes, já que as suas complexidades dependem em certa forma da maneira como tratam a flexibilidade dos pares de proteínas e por consequência do número de dimensões que o espaço solução irá ter assim como a qualidade do conjunto solução encontrado.

## 2.3 BiGGER

O algoritmo BiGGER (acrónimo para *Bimolecular complex Generation with Global Evaluation and Ranking*) [1] de acordo com a classificação referida no ponto 2.2 enquadra-se no paradigma da docagem flexível[2] permite resolver o problema de conseguir prever o ajuntamento dos complexos proteicos, consistindo essencialmente em dois passos, o primeiro efectua uma redução de possíveis configurações resultantes de passos de translação e rotação numa magnitude de cerca de  $10^{15}$  configurações para milhares das anteriores isto através do algoritmo BoGIE relatado no ponto 2.4. A segunda fase do algoritmo consiste em aplicar metodologias de aprendizagem automática de modo a que se possa prever qual das configurações resultantes corresponde ao melhor ajuste entre os dois complexos, isto é, a que tem o score mais elevado. Tal pode ser feito

Em termos de complexidade temporal, este algoritmo assume valores mais optimais ( $O(N^{2.8})$ )[2]) do que os algoritmos que recorrem ao Fast Fourier Transform, estes últimos assumem como complexidade  $O(N^3 * \log_2(N^3))$ [1] e estes por sua vez têm uma melhor complexidade do que a dos algoritmos de correlação espacial ( $O(N^6)$ ). O motivo pelo qual o das 3 espécies de algoritmos o BiGGER assume-se com maior eficiência em termos de computação deve-se ao facto de este ter sido implementado com diversas optimizações face aos algoritmos FFT, uma delas consiste no uso de uma heurística mais eficiente no passo da eliminação de possibilidades: descarta situações em que existem sobreposições entre cores ou até mesmo situações que não cumprem com os limites impostos nas restrições [2].

O tempo de computação, segundo os autores do algoritmo estava situado entre as 2H e as 8H, dependendo do par de proteínas, em contraste com o tempo de execução dos algoritmos FFT que ronda as 6H, com um CPU do ano de 2000 (Intel Pentium II 450 MHz dispõe apenas 1 core)[2]. Segundo a lei de Moore, o número de transistores presentes num CPU duplica a cada 2 anos, e por consequência a capacidade computacional. Num computador actual o tempo de execução do BiGGER provavelmente será menor, demorando entre 1H e 4H por exemplo, e após as paralelizações estarem concluídas espera-se que o



algoritmo consiga efectuar as simulações num prazo de no máximo minutos.

## 2.4 BoGIE

Acrónimo para *Boolean Geometric Interaction Evaluation*[1] [2], é um algoritmo de pesquisa em grelha utilizado na primeira fase do BiGGER, que é referido no ponto seguinte, mais precisamente na amostragem da população total de configurações possíveis para milhares, existem dois processos principais a considerar, sendo o primeiro a definição de uma matriz tridimensional de booleanos em que cada posição da matriz representa uma parcela da forma que o complexo assume, assume o valor 1 se a célula corresponde a uma parcela da proteína cujo centro se encontra a uma distancia tri-dimensional que se designa por esfera de van der waals de qualquer outro átomo pertencente a outra proteína, as células com o valor 0 correspondem a frações do complexo que são consideradas como externas.

O segundo passo gera duas matrizes de booleano semelhantes às anteriores para cada um dos elementos do par: a matriz de superfície (*surface matrix*) e a matriz central (*core matrix*) tal como está ilustrado graficamente na figura 3.1. Os elementos celulares que ocupam a matriz de superfície são aqueles que na matriz inicial do passo anterior assumiram valor 1 mas tinham vizinhos com valor 0, ou seja, pretende-se os pontos de fronteira.

Na segunda matriz constam as células em que quer o seu valor, quer o das suas células vizinhas assumem valor verdadeiro o que corresponde a posições em que o seu centro está próximo do centro do complexo proteico ou podendo até mesmo coincidir. A forma de garantir que se consegue obter a superfície molecular da proteína é através da operação logica XOR (OU exclusivo), que terá como output 1 apenas nos pontos da fronteira, pois é aqui que o resultado do XOR associado aos dois pontos seleccionados dá o valor verdadeiro, já que os valores entre as duas células são diferentes e falso se forem iguais. Sendo assim a complexidade deste algoritmo está associada mais com o primeiro passo do que com o segundo, já que este ultimo depende do output da matriz resultante do primeiro passo e apenas executa um conjunto de operações XOR o que não é assim tão custoso em termos de memória e tempo comparando com a medição para cada célula de uma distancia. De notar, no entanto que ambos os passos podem ser optimizados recorrendo ao GPU, podendo trazer melhorias para além do uso do XOR.

## 2.5 Fast Fourier Transform

## 2.6 Algoritmos de correlação espacial

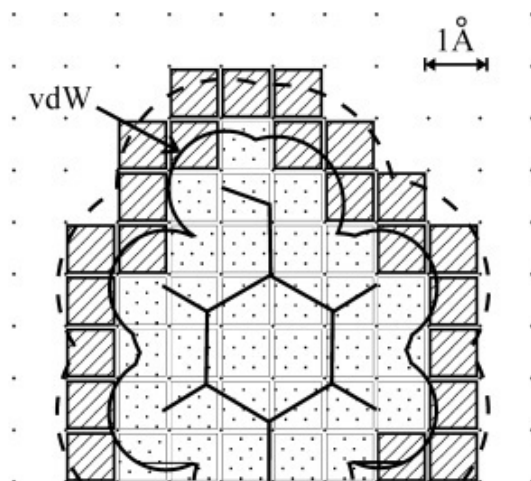


Figura 2.1: Representação em 2D das matrizes resultantes do segundo passo do BoGIE, as células preenchidas a tracejado diagonal correspondem à matriz de superfície, as com pontos correspondem à matriz *core* e a núvem com tracejado contínuo representa o corte associado à esfera de van der waals com a proteína localizada ao centro [2]

## A SHORT L<sup>A</sup>T<sub>E</sub>X TUTORIAL WITH EXAMPLES

This Chapter aims at exemplifying how to do common stuff with L<sup>A</sup>T<sub>E</sub>X. We also show some stuff which is not that common! ;)

Please, use these examples as a starting point, but you should always consider using the *Big Oracle* (aka, [Google](#), your best friend) to search for additional information or alternative ways for achieving similar results.

### 3.1 Document Structure

### 3.2 Dealing with Bibliography

### 3.3 Inserting Tables

### 3.4 Importing Images

### 3.5 Floats, Figures and Captions

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan

eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

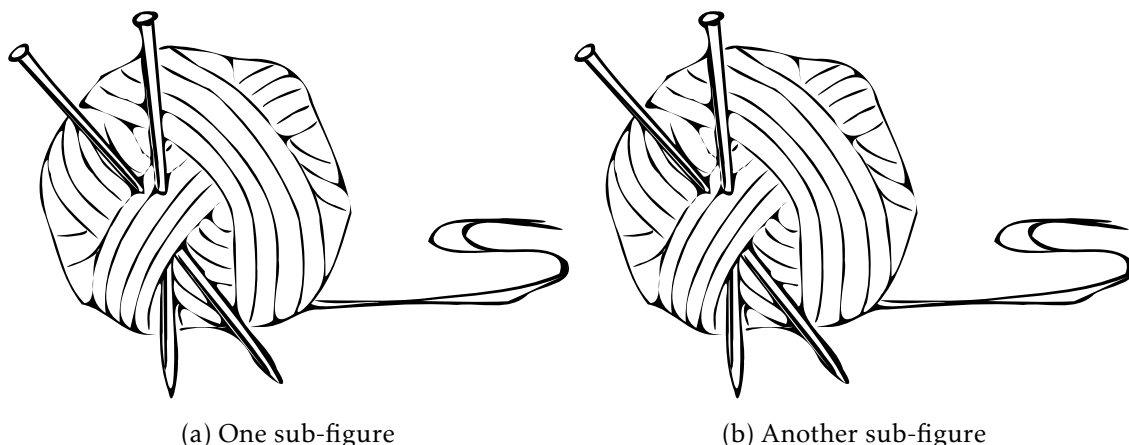


Figura 3.1: A figure with two sub-figures!

**And this is a small text that references the Figure 3.1 and its Subfigures 3.1a and 3.1b.**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan

eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

## 3.6 Text Formatting

## 3.7 Generating PDFs from $\text{\LaTeX}$

### 3.7.1 Generating PDFs with `pdflatex`

You may create PDF files either by using `latex` to generate a DVI file, and then use one of the many DVI-2-PDF converters, such as `dvipdfm`.

Alternatively, you may use `pdflatex`, which will immediately generate a PDF with no intermediate DVI or PS files. In some systems, such as Apple, PDF is already the default format for  $\text{\LaTeX}$ . I strongly recommend you to use this approach, unless you have a very good argument to go for `latex + dvipdfm`.

A typical pass for a document with figures, cross-references and a bibliography would be:

```
$ pdflatex template
$ bibtex template
$ pdflatex template
$ pdflatex template
```

You will notice that there is a new PDF file in the working directory called `template.pdf`. Simple :)

Please note that, to be sure all table of contents, cross-references and bibliographic citations are up-to-date, you must run `latex` once, then `bibtex`, and then `latex` twice.

### 3.7.2 Dealing with Images

You may process the same source files with both `latex` or `pdflatex`. But, if your text include images, you must be careful. `latex` and `pdflatex` accept images in different (exclusive) formats. For `latex` you may use EPS ou PS figures. For `pdflatex` you may use JPG, PNG or PDF figures. I strongly recommend you to use PDF figures in vectorial format (do not use bitmap images unless you have no other choice).

### 3.7.3 Creating Source Files Compatible with both `latex` and `pdflatex`

Do not include the extension of the file in the `\includegraphics` command. E.g., use

```
\includegraphics{sonwman}
```

and not

```
\includegraphics{sonwman.eps}.
```

If you use the first form, `latex` or `pdflatex` will add an appropriate file extension.

This means that, if you plan to use only `pdflatex`, you need only to keep (preferably) a PDF version of all the images. If you plan to use also `latex`, then you also need an EPS version of each image.

## To be included in the sections above

Para fazer citações, deverá usar-se a chave da referência no ficheiro BibTeX. Se for uma única referência [Artho04], usar um “~” para ligar o `\cite{...}` à palavra que o precede (...referência~\cite{Artho04}). Caso queira fazer múltiplas citações [Shavit95, Silberschatz06, Moss85], deverá agrupá-las dentro de um único `\cite{...}`.

Note que o ficheiro de bibliografia pode ter tantas entradas quantas quizer. Apenas aquelas cuja chave seja referenciada no texto é que serão incluídas na listagem de bibliografia.

Footnotes<sup>1</sup> will be numbered and shown in the bottom of the page.

A Tabela 3.1 ilustra alguns conceitos importantes associados à construção de tabelas:

- i) Não usar linhas verticais;
- ii) A legenda deve ficar por cima da tabela;
- iii) Usar as macros `\toprule`, `\midrule` e `\bottomrule` para fazer a linha horizontal superior, interiores e inferior, respectivamente.

Tabela 3.1: Test results summary.

Test	Anomalies	Warnings	Correct	Categories	Missed
[Beckman08] Connection	2	2	1	<i>C</i>	1
[Artho03] Coordinates’03	1	4	1	<i>2B, 1C</i>	0
[Artho03] Local Variable	1	2	1	<i>A</i>	0
[Artho03] NASA	1	1	1	—	0
[Artho04] Coordinates’04	1	4	1	<i>3C</i>	0
[Artho04] Buffer	0	7	0	<i>2A, 1B, 2C, 2D</i>	0
[Artho04] Double-Check	0	2	0	<i>1A, 1B</i>	0
[Flanagan04] StringBuffer	1	0	0	—	1
[Praun03] Account	1	1	1	—	0
[Praun03] Jigsaw	1	2	1	<i>C</i>	0
[Praun03] Over-reporting	0	2	0	<i>1A, 1C</i>	0
[Praun03] Under-reporting	1	1	1	—	0
[IBM-Rep] Allocate Vector	1	2	1	<i>C</i>	0
Knight Moves	1	3	1	<i>2B</i>	0
<b>Total</b>	<b>12</b>	<b>33</b>	<b>10</b>	<b>5A, 6B, 10C, 2D</b>	<b>2</b>

As figuras a inserir no documento deverão ser de qualidade, preferencialmente em formato vectorial (PDF vectorial) e não em *bitmap* (PNG, JPG, etc). As imagens *bitmap* (Figura 3.2) não escalam bem e têm reflexos negativos na qualidade do seu documento. Pelo contrário, as imagens *vectoriais* Figura 3.3 escalam muito tanto quanto o necessário sem degradar a qualidade da imagem.

Só deve usar *screenshots* se não tive mesmo nenhuma alternativa. Em vez de gerar um *screenshot*, tente usar uma impressora virtual PDF e imprimir para um ficheiro PDF. Regra

<sup>1</sup>This is a simple footnote.

geral obterá um PDF vetorial. Mesmo que o seu PDF contenha imagens, elas terão sempre qualidade maior ou igual à que obteria com um *screenshot*.

Para agregar várias figuras numa única... Poderá assim referenciar o conjunto 3.4, a primeira delas 3.4a ou a segunda 3.4b.

Para incluir listagens de código no seu documento, deverá incluir o pacote *listings* e depois usar o ambiente *lstlisting*, como exemplificado na Listagem 3.1.

Listagem 3.1: Hello World

```
1 /**
2  * The HelloWorldApp class implements an application that
3  * simply prints "Hello World!" to standard output.
4  */
5 class HelloWorldApp {%
6     public static void main(String[] args) {%
7         System.out.println("Hello World!"); // Display the string.
8     }
9 }
```

## 3.8 Equações

O LaTeX é uma ferramenta poderosa para escrever em estilo matemático. Permite inserir fórmulas no meio do texto como por exemplo esta:  $ax^2 + bx + c = 0$ . Também permite que as fórmulas sejam destacadas numa linha separada e centradas na página

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

ou numeradas

$$aaa \tag{3.1}$$

que depois pode ser referida no texto como sendo a equação 3.1

$$aa$$

$$a \tag{3.2}$$

$$b \tag{3.3}$$

$$c \tag{3.4}$$

$$\tag{3.5}$$

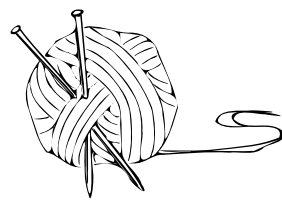




Figura 3.2: Imagem em formato *bitmap* (JPG)



Figura 3.3: Imagem em formato PDF vectorial



a Novelo de lã



b Tempestade com neve

Figura 3.4: Exemplo de utilização de *subbottom*



## BIBLIOGRAFIA

- [1] L. Krippahl. *Integrating protein structural information*. 2003.
- [2] P. N. Palma, L. Krippahl, J. E. Wampler e J. Moura. “BIGGER: A new (soft) docking algorithm for predicting protein interactions”. Em: 39 (jun. de 2000), pp. 372–84.
- [3] H. Wenfan. *Rigid Body Protein Docking by Fast Fourier Transform*. 2005. URL: <https://www.comp.nus.edu.sg/~leowwk/hyp/huangwenfan.pdf>.

