



Universidad Politécnica
de Madrid

**Escuela Técnica Superior de
Ingenieros Informáticos**



Grado en «título del grado»

Trabajo Fin de Grado

**Título del Trabajo, con Mayúscula en
Todas las Palabras que no Sean
Conectivas (Artículos, Preposiciones,
Conjunciones)**

Autor: «nombre y apellidos»

Tutor(a): «nombre y apellidos»

Madrid, «Mes Año»

Este Trabajo Fin de Grado se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Grado

Grado en «título del grado»

Título: Título del Trabajo, con Mayúscula en Todas las Palabras que no Sean Conectivas (Artículos, Preposiciones, Conjunciones)

«Mes Año»

Autor: «nombre y apellidos»

Tutor: «nombre y apellidos»

«departamento»

ETSI Informáticos

Universidad Politécnica de Madrid

Resumen

«Aquí va el resumen del TFG. Extensión máxima 2 páginas.»

Abstract

«Abstract of the Final Degree Project. Maximum length: 2 pages.»

Tabla de contenidos

1. Introducción	1
2. Desarrollo	3
2.1. Conocimientos previos y definiciones	3
2.1.1. Motivación	3
2.1.2. Complejos Simpliciales	4
2.1.3. Homología	4
2.1.4. Persistencia	5
3. Resultados y conclusiones	7
Bibliografía	9
Anexo	10
.1. Ejemplo de código en python	11
.2. Ejemplo de fórmula matemática	11

Capítulo 1

Introducción

LaSSSS introducción del TFG debe servir para que los profesores que evalúan el Trabajo puedan comprender el contexto en el que se realiza el mismo, y los objetivos que se plantean.

Esta plantilla muestra la estructura básica de la memoria final de TFG, así como algunas instrucciones de formato.

El esquema básico de una memoria final de TFG es el siguiente:

- Resumen en español y inglés (máximo 2 páginas cada uno)
- Tabla de contenidos
- Introducción (con los objetivos del TFG)
- Desarrollo
- Resultados y conclusiones
- Bibliografía (publicaciones utilizadas en el estudio y desarrollo del trabajo)
- Anexos (opcional)

En cualquier caso, es el tutor del TFG quien indicará a su estudiante la estructura de memoria final que mejor se ajuste al trabajo desarrollado.

Con respecto al formato, se seguirán las siguientes pautas, que se muestran en esta plantilla:

- *Tamaño de papel:* DIN A4
- *Portada:* tal y como se recoge en esta plantilla, con indicación de universidad, centro, título de TFG y autor.
- *Segunda página:* información bibliográfica, incluyendo todos los datos del tutor del TFG.
- *Tipo de letra para texto.* Preferiblemente “Bookman Old Style” 11 puntos. Si no fuera posible, las alternativas recomendadas son, por orden de preferencia: “Palatino Linotype”, “Garamond” o “Georgia”.
- *Tipo de letra para código fuente:* “Consolas” o “Roboto mono”

-
- *Márgenes*: superior e inferior 3 cm, izquierdo y derecho 2.54 cm.
 - *Secciones y subsecciones*: reseñadas con numeración decimal a continuación del número del capítulo. Ej.: subsecciones 2.3.1.
 - *Números de página*: siempre centrado en margen inferior, página 1 comienza en capítulo 1, todas las secciones anteriores al capítulo 1 en número romano en minúscula (i, ii, iii. . .).

Para elaborar la memoria final del TFG con esta plantilla, seguir los siguientes pasos:

1. Descargar e instalar MiKTeX: <https://miktex.org/>
2. Descargar e instalar un editor de \LaTeX , por ejemplo Texmaker:
<https://www.xmlmath.net/texmaker/>
3. Editar el archivo **secciones/ _DatosTFG.tex**, que hay en la carpeta **secciones** de esta plantilla. Complimentar todos los datos pedidos en dicho archivo. Guardar y cerrar.
4. Compilar el archivo **plantilla_TFG.tex** (puede ser renombrado). Se generará como resultado un archivo **pdf**.
5. Para escribir la memoria final del TFG se pueden añadir y/o modificar los archivos de la carpeta **secciones** como sea necesario. El resultado se obtiene al compilar el archivo **plantilla_TFG.tex**.

Capítulo 2

Desarrollo

2.1. Conocimientos previos y definiciones

En esta sección se mostrarán las principales nociones de Topología Computacional, que nos darán el contexto y conocimientos necesarios para poder comprender el Teorema de Estabilidad y ser capaces de abordar su demostración.

2.1.1. Motivación

La Topología se centra en el estudio de las diversas propiedades de los espacios topológicos y las funciones continuas. Recordemos la definición formal de espacio topológico:

Definición 2.1.1. Sea X un conjunto y $\tau \subset \mathcal{P}(X)$ una colección de de X . Diremos que τ es una topología en X si satisface las siguientes propiedades:

1. \emptyset y X son elementos de τ .
2. Si $\{U_\lambda\}_{\lambda \in \Lambda}$ es una familia de elementos de τ entonces

$$\bigcup_{\lambda \in \Lambda} U_\lambda$$

es un elemento de τ (La unión arbitraria de elementos de τ pertenece τ).

3. Si U_1, \dots, U_n son elementos de τ , entonces

$$U_1 \cap \dots \cap U_n$$

es un elemento de τ (La intersección finita de elementos de τ pertenece τ).

El par (X, τ) se denomina **espacio topológico** y a los elementos de τ se denominan **abiertos** de X .

Mientras que en el subcampo de la Topología Computacional veremos como podemos hacer uso de diversos algoritmos para poder estudiar las propiedades de los espacios topológicos y ser capaces de resolver problemas topológicos computacionalmente. Para ello lo primero que necesitamos es una manera de representación de nuestros espacios topológicos, manteniendo sus propiedades topológicas.

2.1.2. Complejos Simpliciales

Una de las formas de representar un espacio topológico es a través de la descomposición del mismo en piezas más sencillas. Una descomposición en un complejo si sus piezas son topológicamente simples y sus intersecciones son piezas de dimensión inferior del mismo tipo [5]. Dentro de los complejos se puede observar que hay una gran variedad de tipos, dandonos distintos grados de abstracción. Nosotros vamos a trabajar con los complejos simpliciales, ya que nos darán unas buenas capacidades de computación.

Los complejos simpliciales los podemos estudiar desde un enfoque geométrico y desde un enfoque combinatorio. Partiremos de la definición de complejo simplicial desde el punto de vista geométrico.

combinaciones convexas

def simple

caras y cocaras

def complejo simplicial

espacio subyacente y poliedros: compacidad, def triangulación

def subcomplejo

def j-esqueleto

def estrella y link

complejo simplicial abstracto: esquema de vertices, realización geométrica y teorema

def subdivisión de un complejo simplicial

subdivisión baricéntrica: def y lema

aplicaciones simpliciales y aproximaciones simpliciales

Desde el punto de vista computacional nos encontramos con el problema de que tenemos una representación de un espacio topológico a través de una discretización finita de los puntos de dicho espacio, y nuestro objetivo es poder recuperar las propiedades del espacio topológico original a partir de esta nube de puntos. Para ello usaremos complejos simpliciales asociados a dicha nube de puntos.

def complejo de Čech

def complejo de Vietoris-Rips

def diagrama de Voronoi + triangulación Delonay

def alpha complejo

2.1.3. Homología

Como se puede ver en [añadir cita] la homotopía es una gran herramienta algebraica para poder obtener propiedades de los espacios topológicos. Sin embargo, los métodos

para el cálculo de la homotopía no son los mejores computacionalmente. Así pues, se propone la homología como formalismo algebraico, que aunque no es capaz de obtener tanta información topológica sobre el espacio como con otros formalismos, contiene algoritmos mucho más rápidos y eficientes.

grupos de cadenas

operador borde: complejos de cadenas, ciclos y bordes

grupos de homología simplicial: definición, números de betti, característica de euler, teorema de conexión y números de betti

aplicaciones inducidas para grupos de homología

Homología singular

Comentar existencia de la homología singular y el porqué se creó y que beneficios, pero debido a que el teorema de estabilidad parte de un espacio topológico triangulable no es necesario

HERBERT EDELSBRUNNER AND JOHN HARER: PERSISTENT HOMOLOGY — A SURVEY

“REMARK 2.1. There are a variety of other homology theories defined in topology. Most notably singular homology has the advantage that it exists for arbitrary topological spaces and it is easy to define concepts like induced maps, prove that homotopy equivalent maps induce isomorphisms on homology, etc. However, in singular homology the chain groups are infinite-dimensional and therefore not directly suited to computational methods. Nevertheless, the reader should be aware of this theory. It justifies the common practice of talking about homology for spaces without an explicit triangulation. Most of the time, and certainly in low dimensions, singular and simplicial homology are equivalent theories.”

Crossley, Martin D., Essential Topology. Springer-Verlag, London, 2005.

“For simplicial homology we supposed that our space had already been expressed as a simplicial complex, i.e., decomposed into a union of simplices. Some spaces cannot be expressed in such a way, and even those that can, can usually be expressed as a simplicial complex in many different ways. Choosing one way can obscure some details of the space. For these reasons “singular” homology was developed, which gets around this problem by, in a very loose sense, considering all possible simplicial decompositions.”

2.1.4. Persistencia

funciones reales

poder ver gráficamente como a través de las contraímagenes se generan diferentes componentes al pasar por puntos críticos

funciones morse

de variedad diferenciable, definición de las funciones morse, ver como la homología cambia al pasar por los puntos críticos y ver las nociones de nacimiento y muerte

funciones tame

definición función tame, subniveles, grupos de persistencia a partir de funciones tame, critical value lema, numeros de betti de grupos de persistencia y su relación con las multiplicidades.

Definición de diagrama de persistencia, k-triangle lemma

persistencia en complejos simpliciales

particularizar lo visto en funciones tame pero con complejos simpliciales

funciones PL

definición de las funciones pl, generación de filtraciones a partir del lower star filtration y estudio de los puntos criticos de las funciones pl para comprobar que son funciones tame.

Capítulo 3

Resultados y conclusiones

Resumen de resultados obtenidos en el TFG. Y conclusiones personales del estudiante sobre el trabajo realizado.

Bibliografía

- [1] Publicaciones utilizadas en el estudio y desarrollo del trabajo. Hay que utilizar un sistema internacional para referencias bibliográficas, de acuerdo con las indicaciones del tutor. Por ejemplo, el **sistema de IEEE**.
- [2] M. de Guzmán y B. Rubio, *Integración: Teoría y Técnicas*, Alhambra, Madrid, 1979.
- [3] P. Mattila, *Geometry of Sets and Measures in Euclidean Spaces*, Cambridge University Press, Cambridge, 1995.
- [4] C.A. Rogers, *Hausdorff Measures*, Cambridge University Press, Cambridge, 1998.
- [5] H. Edelsbrunner y J. Harer, *Computational Topology: An Introduction*, 2010.

Anexo

Este capítulo es opcional, y se escribirá de acuerdo con las indicaciones del Tutor.

.1. Ejemplo de código en python

```
1 # -*- coding: utf-8 -*-  
2 import sympy as sy  
3 from sympy.abc import x
```

.2. Ejemplo de fórmula matemática

$$e^{i\pi} + 1 = 0$$