Grau en Matemàtica Computacional

Estada en Pràctiques i Projecte Final de Grau

Semigrups numèrics: semimòduls, sizígies i la seua relació amb la conjetura de Wilf

Autor:
Alfredo Granell Marqués

Supervisor:
Aaron Martínez Romero
Tutor acadèmic:
Julio José Moyano Fernández

Data de lectura: __ de ________ de 20__
Curs acadèmic 2019/2020

Resum

Aquest document recull l'estada en pràctiques i el projecte final de grau de l'assignatura MT1030 – Pràctiques Externes i Projecte de Final de Grau del grau en Matemàtica Computacional en dues parts diferenciades.

En la primera part del treball es descriu l'estada en pràctiques en l'empresa *ArkerLabs* on s'ha aplicat la teoria de cues per desenvolupar un algoritme d'estimació del temps d'espera en una cua per implementar-lo en el programari *Inqueue*.

En la segona part del treball, estudiem les generalitats i resultats més bàsics dels semigrups numèrics. A, més, descrivim els conceptes de semimódul, sizígia i la seua relació per a semigrups numèrics generats minimalment per dos elements. Finalment, generalitzem aquests conceptes i, a través de proves numèriques calculades per programes dissenyats per a aquest treball, donem alguns resultats que relacionen la conjectura de Wilf amb l'estudi dels semimóduls de certs semigrups numèrics.

Paraules clau

semigrup numèric, semimódul, sizígia, conjectura de Wilf

Abstract

This document presents the external work placement and bachelor's degree final project from the subject MT1030 – External Work Placement and Bachelor's Degree Final Project from the bachelo's degree in Computational Mathematics in two differentiated parts.

In the first part of this paper, it is explained the external work placement at the company *ArkerLabs*. During this external work placement, it was used the queue theory in order to develop an algorithm to estimate the waiting time in a queue to implement itself in the software *Inqueue*.

En the second part of this paper, we study the basic concepts and most important results regarding the numerical semigroups. Furthermore, we describe the concepts of semimodule, syzygy and their relationship in numerical semigroups minimally generated by two elements. Finally, we generalize these concepts and, through numerical tests given by programmes designed for this paper, we present some results which relate the Wilf's conjecture with the study of semimodules of some numerical semigroups.

Keywords

numerical semigroup, semimodule, syzygy, Wilf's conjecture

Índex

Part I

Estada en pràctiques

Capítol 1

Descripció de la empresa

ArkerLabs és una empresa de desenvolupament de programari especialitzada en tecnologia *blockchain*. Els serveis que ofereix l'empresa van més enllà de la creació i manteniment de projectes basats en tecnologia *blockchain*, ja que també realitzen solucions de programari personalitzades per a tota classe de clients. A més, l'empresa també funciona com a consultoria per assessorar en qualsevol projecte informàtic que presente un client, especialment en projectes que implementen tecnologia *blockchain*.

Alguns dels productes que ofereixen són *Arker: La leyenda de Ohm*, un videojoc que crea la seua pròpia economia dins del joc utilitzant una criptomoneda; *Kashless*, una solució per al control d'accesos i el pagament en efectiu aplicable a esdeveniments o comerços i *Kusari*, una *blockchain* permmisioanda dirigida a l'emmagatzematge i registre de dades de manera distribuïda.

La seua oficina central està ubicada en l'edifici Espaitec 1 al Parc Científic, Tecnològic i Empresarial de la Universitat Jaume I a Castelló de la Plana. L'empresa compta amb nou empleats dels quals Juanjo Chust i Aaron Martínez són co-CEO i a més CTO i COO respectivament. Nelson Marco és DevsOps i Edu Simon és Project Manager mentre que el gruix productor de l'empresa el formen Marc Jovaní, Sergio Montañes i Jorge Martínez com a Full Stack Developers i Carlos Chust com a Junior Developer. Finalment, Mahuay Fernandez realitza les labors de Community Manager.

L'estada s'ha desenvolupat en l'àrea d'innovació i desenvolupament de nou programari. Aquest departament s'encarrega del disseny i la programació de noves solucions basades en programari. El treballador de l'empresa més relacionat amb l'estada ha sigut el supervisor de les pràctiques Aaron Martínez però han col·laborat en major o menor grau una part important de l'equip aportant retroacció en les diferents fases del projecte o ajudant en la neteja del codi.

Capítol 2

Objetius del projecte formatiu

A un nivell molt general, l'objectiu principal de l'estada en pràctiques és obtindre de primera mà una experiència professional relacionada amb la temàtica i els continguts oferits en el grau en Matemàtica Computacional. D'aquesta manera, s'ofereix una oportunitat per a introduir-se en el món laboral amb garanties de realitzar un treball relacionat amb la informàtica i les matemàtiques. Gràcies a aquest mecanisme, s'obtindrà una formació completa que integrarà els coneixements assolits a l'aula amb experiència laboral en l'àmbit del grau.

Des del punt de vista més acadèmic, es pretén que el projecte formatiu facilite l'aprenentatge autònom de nous coneixements informàtics i tècniques d'aplicació de les matemàtiques a situacions reals. Així doncs, s'ampliaran les nocions adquirides en altres matèries del pla d'estudis. A més, també es pretén que s'apliquen aquells coneixements adquirits en la titulació de forma pràctica que ajudarà a assolir-los i augmentar-los.

Per altra part, l'estada també té objectius de naturalesa més professional. En aquest sentit, un objectiu important és conéixer el funcionament intern d'una empresa dedicada a la informàtica i ser capaç d'integrar-se en la seua estructura realitzant un treball relacionat amb la matemàtica computacional. També es pretén que es planifique el treball realitzat i es documente detalladament per a la posterior redacció d'una memòria tanmateix com poder presentar els resultats obtinguts a la mateixa empresa.

Fins i tot, també és objectiu del projecte formatiu desenvolupar habilitats transversals essencials en el món laboral com poden ser la resolució de problemes, la preocupació per la qualitat i la capacitat de generar noves idees.

Capítol 3

Explicació detallada del projecte realitzat a l'empresa

3.1 Metodologia i definició de tasques

Inqueue és un programari per solucionar l'espera presencial en una cua per a comerços i esdeveniments. De cara a l'usuari, permet obtindre un torn d'accés mitjançant una aplicació i mostra un temps d'espera aproximat en la cua. De cara al client, mostra un informe amb dades rellevants per a conéixer les característiques del servei que ofereix i optimitzar els seus recursos.

El projecte *Inqueue* es trobava en una fase primerenca del seu desenvolupament però, en el moment de començar l'estada, ja s'havia realitzat una prova d'ús real de l'aplicació. Per a aquesta prova, s'havia desenvolupat una aplicació per a telèfons intel·ligents on els usuaris d'un comerç agafaven torn en la cua digitalment i se'ls presentava un compte arrere que representava el seu temps d'espera. Els resultats de la prova no van ser satisfactoris per la imprecisió de l'algoritme implementat en el càlcul del temps d'espera. Es va decidir prescindir d'aquest algoritme i *Inqueue* es va redissenyar quasi totalment.

En aquest context es van definir les tasques del projecte formatiu. Inicialment, les tasques proposades foren l'estudi de la teoria de cues i la seua aplicació en *Inqueue* així com el disseny de diferents algoritmes per a estimar el temps d'espera per a diferents casos d'ús. A més, els algoritmes s'haurien d'implementar en un banc de proves proporcionat per l'empresa.

A causa de circumstàncies relacionades amb el projecte, algunes d'aquestes tasques es modificaren al llarg de l'estada. L'aplicació de la teoria de cues bàsica, que és la que es pretenia estudiar, no va resultar útil al projecte perquè els temps d'espera aproximats que proporcionava aquesta teoria eren menys precisos del que es desitjava. A més, s'arribà a la conclusió que es podia proporcionar la millor estimació de temps d'espera amb un únic algoritme. Malauradament, com *Inqueue* s'estava redissenyant quasi per complet, tampoc es va proporcionar un banc de proves per comprovar l'eficàcia de l'algoritme proposat.

Conseqüentment, les tasques proposades per a la seua finalització foren l'estudi de la teoria de cues, el disseny d'un algoritme que estimara el temps d'espera en una cua amb poc marge d'error i el desenvolupament d'un banc de proves propi.

3.2 Planificació temporal de les tasques

L'estada en pràctiques començà el dia 3 de febrer de 2020 i acabà el dia 7 de maig de 2020. La durada total de l'estada ha sigut de 290 hores amb horari de 9.00 a 14.00 de dilluns a divendres. La planificació inicial fou la següent: dissenyar diferents algoritmes d'estimació de temps al llarg del mes de febrer i al llarg de març i abril, modelitzar les cues i generar un resum de dades importants sobre el funcionament del sistema de cues per al client. Finalment, s'invertiria el temps restant de pràctiques en implementar els algoritmes en la plataforma que menejarà l'aplicació utilitzant el llenguatge de programació TypeScript.

Tal com s'ha indicat a la Secció 3.1, la planificació inicial es va veure altament alterada i modificada en repetides ocasions al llarg de l'estada. Per aquest motiu, s'indica a continuació el treball planificat per a cada setmana, el treball realment realitzat i les modificacions pertinents a les tasques definides.

3.2.1 1a setmana

La primera tasca a realitzar era el recull d'informació al voltant de la teoria de cues i familiaritzar-se amb aquesta. Mitjançant una cerca superficial, s'obtingué coneixement dels sistemes de cues modelitzats més bàsics que es denoten seguint la notació de $Kendall$. La versió més simple d'aquesta notació és de la forma $A/S/s$ on A i S denoten la distribució de probabilitat del temps entre arribades consecutives d'usuaris a la cua i del temps de servei a cada usuari respectivament. s denota el nombre de servidors treballant en paral·lel. A més, per a aquests sistemes existeixen les fórmules de *Little*, que permeten calcular diferents paràmetres del sistema incloent el temps d'espera mitjà en la cua. Així doncs, el següent pas fou la implementació de les fórmules de *Little* en sistemes $M/M/1$, $M/M/s$ i $M/G/1$ en Python.[1]

Amb les fórmules de *Little* implementades, es demanà ajuda a María Victoria Ibáñez Gual,

[1] M denota la distribució exponencial i G una distribució qualsevol

del departament d'estadística i investigació operativa, per aclarir alguns conceptes de teoria de cues. Gràcies a aquesta tutoria, s'arribà a un nou algoritme vàlid per a totes les situacions. Bàsicament, consistia a calcular el temps de servei mitjà dels servidors i multiplicar-lo pel nombre de persones en cua davant de l'usuari per obtindre el seu temps aproximat d'espera en cua. Per últim, les indicacions de María Victoria foren el recull de dades reals per a provar l'eficàcia de l'algoritme implementat.

3.2.2 2a setmana

En aquest punt del projecte, es va decidir dividir el problema en dues parts molt diferenciades. Per una banda, la implementació de l'algoritme i d'altra modelitzar el sistema utilitzant la teoria de cues i generar dades rellevants per al client de cara a la seua optimització. S'optà per encarregar-se primer d'implementar l'algoritme.

En un primer moment, s'intentà generar dos conjunts de dades generats a partir de la distribució exponencial perquè feren el paper dels temps entre arribades i dels temps de serveis. Aquesta aproximació no va resultar satisfactòria perquè els dos conjunts havien de ser coherents: mentre tots els servidors estigueren ocupats, no podia accedir al servei cap altre usuari. Per tant, es demanà a l'empresa les dades d'una prova de concepte que havien realitzat fa uns mesos on es registraven els temps quan un usuari demanava torn, començava a ser atés i eixia del sistema. Aquestes dades es trobaven en una API implementada sobre *Firebase*. Després de les explicacions d'Edu Simón, es programà en Python un accés a aquestes dades. El següent pas fou organitzar totes aquestes dades en una estructura per a poder treballar sobre elles. Es trià l'estructura d'un *DataFrame* fent servir el paquet *Pandas* per la facilitat a l'hora de poder seleccionar determinades files o columnes.

3.2.3 3a setmana

Una vegada hi havia un conjunt de dades, es va procedir a la implementació de l'algoritme. Es calculava per a cada usuari la seua aproximació de temps i s'obtenia de les mateixes dades el temps real que estigué esperant. Al veure la gran diferència entre els dos resultats, s'optà per demanar una nova tutoria a María Victoria. Fruit d'aquesta reunió, es millorà l'algoritme tenint en compte ara els possibles abandons d'usuaris de la cua.

Malauradament, les condicions en les quals es realitzà la recollida de dades no foren molt òptimes. Tant els usuaris com els mateixos servidors no comprengueren massa bé el funcionament de la plataforma i per tant les dades no eren fiables per a traure conclusions al respecte. Per això, es va cercar les principals fonts de bases de dades per veure si existien registres que serviren per a posar en pràctica l'algoritme. Com no es va poder trobar una base de dades adequada, es passà a intentar simular les dades una vegada més, però aquesta vegada utilitzant un paquet de simulació anomenat *SimPy*.

3.2.4 4a setmana

Es va concertar una reunió per al dia 29 de febrer per presentar els avanços en el projecte *Inqueue* i es dedicà tota aquesta setmana a la preparació d'una exposició oral a tots els implicats en aquest projecte. Es preparà tot el material necessari per a la presentació que estava formada per tres parts.

Primerament, es presentaren els conceptes bàsics d'estadística, i més concretament de teoria de cues, per fer a l'audiència comprendre el context del problema utilitzant les dades recollides per exemplificar els conceptes. Seguidament, es presentaren dues versions de l'algoritme d'estimació del temps d'espera: una primera versió més senzilla i una segona que intentava predir en major exactitud el temps d'espera però que també presentava una major complexitat. Finalment, es proposà la modelització dels sistemes de cues com a font d'informació de cara a presentar dades als clients informant-los del rendiment dels seus sistemes de servei.

3.2.5 5a setmana

Gràcies als comentaris rebuts en la presentació, es van definir noves tasques i perfilades algunes altres. En primer lloc, es va decidir utilitzar l'algoritme més simple però en algunes millores com eliminar dades atípiques al calcular la mitja del temps de servei de cada servidor i realitzar una mitja entre tots els servidors en cas que hi haguera més d'un. També es va decidir que la tasca més prioritària era desenvolupar l'algoritme d'estimació de temps d'espera i que la part de presentar informació al client seria una preocupació per més endavant.

Així doncs, una vegada finalitzat l'algoritme, calia dissenyar una base de dades que permetera un càlcul ràpid dels paràmetres necessaris en el càlcul del temps d'espera. Es va decidir que el sistema a utilitzar seria *MongoDB* per aquest motiu i la familiaritat de l'empresa amb ell. També es decidiren quines dades eren necessàries emmagatzemar en la base de dades i es dissenyaren quines consultes obtindrien aquests paràmetres necessaris.

3.2.6 6a setmana

Finalment, va començar la implementació de la rutina que, a partir d'uns paràmetres prèviament calculats en la base de dades, donava com a resultat el temps d'espera aproximat de l'usuari en la cua. Aquesta rutina es va implementar en *TypeScript* perquè la resta del programari de *Inqueue* també estaria implementada en aquest mateix llenguatge. Es va procedir a escriure aquesta rutina i, una vegada acabada, a netejar el codi reescrivint el nom de les variables perquè foren més explicatives i eliminant línies innecessàries.

A més, com no es podia avançar més mentre altra part de l'equip acabara la seua part, es va escriure un informe on s'explicà en tot detall l'algoritme d'estimació de temps d'espera amb possibles millores i ajustos que devien realitzar-se una vegada provat en una situació real. També s'inclogué en aquest document quines dades havien de recollir-se i de quina manera es devien gestionar per a obtindre els paràmetres necessaris en la rutina programada, que també estava present en l'informe. Aquest document es pot trobar adjunt (en castellà) a l'Apèndix A.

3.2.7 Resta de l'estada

El 14 de març de 2020 es va declarar a l'estat espanyol l'estat d'alarma. Per culpa de la situació de confinament viscuda des d'aquell moment fins al 21 de juny, l'empresa va paralitzar el projecte *Inqueue* i no va ser possible continuar treballant en ell.

Quan acabà l'estat d'alarma i es va decidir amb continuar gradualment amb aquest projecte, l'estada en pràctiques ja havia acabat. Des del 16 de març fins al 7 de maig, es va estar de forma telemàtica a l'espera de noves instruccions per part de l'empresa.

3.3 Recursos del projecte

3.3.1 Teoria de cues

És molt comú que en el món actual en el qual vivim es formen línies d'espera o cues. Açò ocorre quan hi existeix una demanda per un servei és més elevada que la capacitat del sistema per oferir aquest servei. És més, les cues són un fenomen que es donen molt sovint en el context de la informàtica. En aquestes i més situacions es donen unes característiques comunes que permeten la modelització d'un sistema d'una cua i donen lloc a la teoria de cues.

Pel plantejament del problema, eren necessaris coneixements bàsics al voltant de la teoria de cues. Es va estudiar la descripció general d'un sistema en una cua i la notació més utilitzada per a representar sistemes d'una cua: la notació de Kendall. També es van repassar els coneixements relacionats amb les distribucions més populars obtinguts al llarg del grau. Principalment, s'han utilitzat els capítols 10 i 11 de [1] per a l'estudi de la teoria de cues.

3.3.2 Python

Python és un llenguatge de programació interpretat i orientat a objectes. Les seues característiques principals són la bona legibilitat del seu codi i la seua simplesa sense perdre potència o velocitat. Més concretament, la versió utilitzada en el projecte ha sigut *Python 3.6* en l'entorn de programació *PyCharm*.

En primer lloc, es repassaren tots els conceptes generals de la programació en *Python* donats al llarg de tot el grau. En segon lloc, s'estudiaren els mètodes per a connectar-se a través de *Python* a un entorn en *Firebase*, plataforma de desenvolupament d'aplicacions on estaven registrades les dades d'una prova de camp. A més, s'hagué d'aprendre com guardar aquestes dades en una estructura adequada, organitzar-les i treballar en elles a través del paquet *Pandas* recorrent a [2, DataFrame].

3.3.3 MongoDB

MongoDB és una base de dades distribuïda, basada en documents i d'ús general dissenyada especialment per a treballar en el núvol. La seua característica més diferencial és el disseny basat en documents JSON que aproxima més el treball de base de dades a la programació orientada a objectes a diferència de l'aproximació més tradicional basada en files i columnes.

Es van repassar els conceptes generals de les bases de dades presents en el pla d'estudis i s'estudiaren les particularitats del sistema no relacional. S'utilitzà principalment [3, Introduction to MongoDB] per estudiar el sistema de base de dades *MongoDB*.

3.3.4 TypeScript

TypeScript és un llenguatge de programació de codi obert construït sobre JavaScript, un dels llenguatges de programació més utilitzats. *TypeScript* estén a *JavaScript* afegint-li definicions de tipus estàtics. D'aquesta manera, s'ofereix una descripció millor dels objectes amb una documentació millorada.

Primerament, s'estudiaren els conceptes més bàsics presents en [4, Basic Types, Functions, Classes] i en acabar el codi, s'estudiaren bones pràctiques de neteja de codi a partir de [5, StyleGuide].

3.4 Grau de consecució de les tasques proposades

Com ja s'ha comentat en la Secció 3.1 i justificat en la Secció 3.2, la definició de tasques inicial va variar i per tant es discutirà el grau de consecució de les tasques finalment proposades. Cal recordar que aquestes tasques foren l'estudi de la teoria de cues i més concretament tots aquells resultats relacionats amb el temps d'espera, el disseny d'un algoritme que estimara el temps d'espera en una cua que resultara satisfactori per a l'usuari i el desenvolupament d'un banc de proves propi.

Respecte a l'estudi de la teoria de cues, s'han assolit els conceptes més bàsics d'aquesta com ara la notació de Kendall i l'estudi dels models més simples de sistemes de cues, aquells en els que el temps entre arribades consecutives d'usuaris i el temps dels serveis prestats pels servidors es distribueixen segons la distribució exponencial. També s'han estudiat i inclús implementat les famoses fórmules de Little on una invariant que es pot calcular és el temps d'espera mitjà d'un usuari en el sistema.

Pel que fa a l'algoritme, es va arribar a dissenyar un algoritme únic per a calcular el temps d'espera aproximat en una cua, d'acord amb la seua implementació en *Inqueue*. Aquest disseny es va plasmar en un informe, adjuntat en l'Apèndix A, presentat a l'empresa on s'especifiquen els camps de la base de dades que són necessaris per al càlcul de l'aproximació, com calcular els tres paràmetres essencials en el càlcul i finalment què operacions realitzar amb els paràmetres. A més, també s'afig un apartat amb possibles millores i ajustos per fer més precisa l'estimació.

Finalment, en relació a la tasca de desenvolupar un banc de proves propi, es van implementar en *Python* diferents programes que simulaven el registre de dades propi d'un sistema de cues amb les marques de temps d'arribada de cada usuari, quan començava a rebre servei i quan deixava el sistema.

3.5 Conclusions

Pel que fa a les tasques proposades i la consecució d'aquestes descrites en la Secció 3.4, el treball realitzat no ha sigut molt satisfactori. L'estudi de la teoria de cues ha sigut prou superficial i finalment l'algoritme, que devia basar-se en gran part dels coneixements teòrics d'aquesta, ha implementat pocs continguts teòrics. No obstant, sí que ha resultat útil aquest estudi per a conéixer de primera mà l'aplicació d'aquesta teoria en un problema real. Tampoc s'han obtingut els resultats esperats al desenvolupar un banc de proves. Ha resultat la tasca més complicada de totes i encara que s'ha aplicat l'algoritme en alguns programes propis que simulaven el comportament d'una cua, la fiabilitat dels resultats no és molt alta.

Respecte al disseny de l'algoritme d'estimació de temps d'espera en una cua, s'ha arribat a un punt que ha resultat satisfactori. S'han estudiat diferents formes de resoldre el problema presentat i s'ha anat construint tot un procés molt detallat de com s'hauria de captar les dades, com analitzar-les i produir el resultat desitjat a més d'adaptar-se al context concret on s'implementaria. En canvi, el bo o no que és l'aproximació és quasi pràcticament teòric perquè no s'ha pogut implementar en un cas real d'ús.

Personalment, trobe que l'estada en pràctiques no ha sigut útil o productiva respecte a les tasques proposades perquè, com s'ha mencionat just abans, la producció de les tasques proposades no ha sigut l'esperada. El que devia ser un gran programa implementat dins d'una plataforma més gran encara ha acabat sent un document on s'explica com hauria de resoldre's el problema en compte d'implementar una solució. La veritat és que les pràctiques no han sigut satisfactòries en aquest sentit en gran part perquè l'empresa no ha desenvolupat el projecte *Inqueue* al mateix ritme que el treball que he realitzat i s'ha arribat fins on s'ha pogut.

Malgrat les males sensacions en aquest sentit, sí que trobe que les pràctiques han sigut realment útils en altres aspectes. És fàcil de veure llegint les seccions anteriors que incialment es proposà un treball que no s'havia pensat massa bé. Realment, el que ha passat és que se'm presentà un problema a resoldre i he hagut d'anar investigant com fer-lo, resolguen els problemes previs que anava trobant. Al llarg de tota l'estada, he hagut d'aprendre llenguatges de programació nous, dissenyar bases de dades i inclús pensar en la part comercial del projecte i presentar les meues propostes als inversors. En aquest sentit, he desenvolupat molts coneixements transversals i especialment la capacitat de poder trobar una solució a un problema molt obert, una capacitat que s'assegurava des de la promoció del grau en Matemàtica Computacional que arribaríem a desenvolupar enormement.

Part II

Memòria TFG

Capítol 4

Motivació y Objectius

L'estudi dels semigrups numèrics és equivalent a les solucions d'enters no negatius d'una equació lineal no homogènia amb coeficients enters positius que tenen màxim comú divisor u. Ja al final del segle XIX, matemàtics com Sylvester o Frobenius treballaren en aquest camp i fins i tot Frobenius proposà el problema de donar una fórmula per al major enter que no es puga representar com una combinació lineal amb coeficients enters no negatius d'un conjunt d'enters positius donats amb màxim comú divisor u. Aquest problema es coneix com el Problema de Frobenius.

Igual que en l'estructura algebraica d'anell, en el món dels semigrups existeixen els ideals i ideals fraccionaris. Aquests últims es coneixen en el context dels semigrups numèrics com ideals relatius o semimóduls, volent dir que tenen un anàleg de l'estructura de módul sobre un semigrup. A més, per a cada semimódul es pot obtindre un semimódul associat al qual anomenem sizígia.

En el present treball es dona una vista general al voltant dels semigrups numèrics, en especial atenció a les seues invariants més importants, per a després introduir els conceptes tant de semimódul com de sizígia i, a partir de resultats coneguts per a casos particulars de semigrups numèrics, ajudar en l'estudi de la conjectura de Wilf.

Capítol 5

Generalitats dels semigrups numèrics

Al llarg de tot el capítol revisarem els resultats més bàsics per entendre el món dels semigrups numèrics, partint de l'estructura de semigrup, i presentant les invariants més importants d'aquest món. Utilitzarem les seccions 1, 2, 3 de [6, Chapter 1] com a punt de partida seleccionant la informació que ens parega més rellevant per al nostre treball. També recorrerem a [7] quan parlem del conjunt d'Apéry. A més, explicarem en més detall alguns resultats i demostracions per afavorir la comprensió del text amb menys exigència de coneixements matemàtics en general i d'àlgebra en particular.

5.1 Monoides i semigrups numèrics

Un *semigrup* és un conjunt H junt amb una operació binària $+$ tal que $+$ és tancada en H i associativa. És a dir, es compleix que per a qualssevol $a, b \in H$, es té que $a + b \in H$ i sent $c \in H$ un altre element, es dona la igualtat $(a + b) + c = a + (b + c)$.

Un *monoide* és un semigrup $(H, +)$ amb element neutre per a l'operació $+$. Això és, que existeix un element $0 \in H$ tal que donat $a \in H$ qualsevol, es té que $0 + a = 0$ i $a + 0 = 0$. Si a més, l'operació $+$ és commutativa ($a + b = b + a$ per a qualssevol $a, b \in H$), el monoide $(M, +)$ s'anomena *monoide commutatiu*.

Exemple 5.1.1. Si $\mathbb{N} = \{z \in \mathbb{Z} \mid z \geq 0\}$, $(\mathbb{N}, +)$ és un monoide commutatiu.

Un *submonoide* del monoide $(M, +)$ és un subconjunt $N \subseteq M$ tal que hereta del monoide

$(M, +)$ l'estructura de semigrup respecte de l'operació $+$ (tancada i associativa en N) i $0 \in N$. Cal destacar que $\{0\}$ és un submonoide del monoide $(M, +)$ anomenat *submonoide trivial*. La intersecció de submonoides d'un monoide $(M, +)$ és, una vegada més, un submonoide. A més, donat un submonoide $(N, +)$ i un conjunt $A \subseteq M$, el submonoide més menut del monoide $(M, +)$ que conté a A és

$$\langle A \rangle = \{\lambda_1 a_1 + \cdots + \lambda_n a_n \mid \lambda_1, \ldots, \lambda_n \in \mathbb{N} \text{ i } a_1, \ldots, a_n \in A\}.$$

Anomenem a aquest submonoide com al *submonoide generat per A* i al conjunt A com al seu *sistema de generadors*. El monoide $(M, +)$ estarà *finitament generat* si existeix un sistema de generadors finit.

Un semigrup numèric $(S, +)$ és un submonoide del monoide $(\mathbb{N}, +)$ tal que el seu complement és finit, és a dir, que $\mid \mathbb{N} \setminus S \mid < +\infty$. Per afavorir la lectura del treball, d'ara endavant quan mencionem un monoide o semigrup numèric, ometrem l'operació binària $+$ escrivint S en compte de $(S, +)$.

Vegem a continuació una caracterització molt potent dels semigrups numèrics. Aquest resultat ens permet comprovar, a partir del conjunt de generadors, si es té un semigrup numèric o no.

Proposició 5.1.2 ([6], Lemma 2.1)**.** *Siga A un subconjunt no buit de $\mathbb{N}$. Aleshores $\langle A \rangle$ és un semigrup numèric si i sols si el màxim comú divisor de A és 1, és a dir, que* $\operatorname{mcd}(A) = 1$.

Demostració. Siga $\langle A \rangle$ un semigrup numèric i $d = \operatorname{mcd}(A)$. Siga $s \in \langle A \rangle$, existiran $\lambda_i \in \mathbb{N}$, $a_i \in A$ tals que $s = \lambda_1 a_1 + \cdots + \lambda_n a_n$ per a $i \in \{1, \ldots, n\}$. Aleshores, com $d \mid a_i$ per a $i \in \{1, \ldots, n\}$, tindrem que $d \mid s$. Com $\langle A \rangle$ és un semigrup numèric, $\mathbb{N} \setminus \langle A \rangle$ és finit i per tant existirà $s \in \langle A \rangle$ tal que $s + 1 \in \langle A \rangle$ perquè $\langle A \rangle$ és infinit. Així doncs, $d \mid s$ però també $d \mid (s+1)$ llavors $d = 1$.

Siga $\operatorname{mcd}(A) = 1$. Per la identitat de Bézout, existiran $z_1, \ldots, z_n \in \mathbb{Z}$ i $a_1, \ldots, a_n \in A$ tals que $z_1 a_1, \ldots, z_n a_n = 1$. Si passem els elements z_i negatius al membre de la dreta, podem trobar $i_1, \ldots, i_k, j_1, \ldots, j_l \in \{1, \ldots, n\}$ tals que $z_{i_1} a_{i_1} + \cdots + z_{i_k} a_{i_k} = 1 - z_{j_1} a_{j_1} - \cdots - z_{j_l} a_{j_l}$. Per tant, si denotem $s = -z_{j_1} a_{j_1} - \cdots - z_{j_l} a_{j_l} \in \langle A \rangle$, tenim que existeix $s \in \langle A \rangle$ tal que $s + 1 \in \langle A \rangle$. Siga $n \in \mathbb{N}$, provem que si $n \geq (s-1)s + (s-1)$, aleshores $n \in \langle A \rangle$. Utilitzant l'algoritme de divisió, tenim que $n = qs + r$ amb $q, r \in \mathbb{N}$ i $0 \leq r < s$. Per tant,

$$n = qs + r \Leftrightarrow qs = n - r \geq (s-1)s + (s-1) - r > (s-1)s + (s-1) - s = (s-1)s - 1.$$

És a dir, que

$$qs > (s-1)s - 1 \Leftrightarrow qs \geq (s-1)s \Leftrightarrow q \geq s - 1.$$

Per tant, tenim que $q \geq s - 1 \geq r$ i finalment obtenim que

$$n = qs + r = qs + r + rs - rs = r(s+1) + (q-r)s \in \langle A \rangle.$$

Així doncs, a partir de l'element $(s-1)s+(s-1)$, tots els elements de $\mathbb{N}$ pertanyen a $\langle A\rangle$ provant que el conjunt d'elements que poden pertànyer a $\mathbb{N}\setminus A$ és finit i conseqüentment $\mathbb{N}\setminus A$ és finit. $\square$

Donats dos monoides M i N, l'aplicació $f\colon M\mapsto N$ és un *homorfisme de monoides* si $f(a+b)=f(a)+f(b)$ per a qualssevol $a,b\in M$ i $f(0)=0$. Diguem que f és un *monomorfisme* o un *epimorfisme* si f és injectiva o suprajectiva respectivament. Si l'aplicació f és bijectiva (injectiva i suprajectiva) diguem que f és isomorfisme i que M i N són isomòrfics. Denotem aquest fet com a $M\cong N$.

Proposició 5.1.3 ([6], Proposition 2.2)**.** *Siga M un submonoide no trivial de $\mathbb{N}$. Aleshores M és isomòrfic a un semigrup numèric.*

Demostració. Siga $d=\mathrm{mcd}(M)$ i $A:=\{\frac{m}{d}\mid m\in M\}$. Com $\mathrm{mcd}(A)=1$, per la Proposició 5.1.2, tenim que $S=\langle A\rangle$ és un semigrup numèric. Vegem que l'aplicació $f\colon M\mapsto S$, $f(m)=\frac{m}{d}$ és un isomorfisme de monoides. Siguen $a,b\in M$, tenim que

$$f(a)+f(b)=\frac{a}{d}+\frac{b}{d}=\frac{a+b}{d}=f(a+b)$$

i sent $0\in M$, tenim que

$$f(0)=\frac{0}{d}=0\in S.$$

Açò prova que f és un homomorfisme de monoides. Si $f(a)=f(b)$, aleshores

$$\frac{a}{d}=\frac{b}{d}\Leftrightarrow a=b.$$

A més, per a tot $s\in S$, existeix m tal que $s=\frac{m}{d}$. Queda provada així la injectivitat i suprajectivitat respectivament. Per tant, f és un isomorfime de monoides. $\square$

Aquest resultat mostra que, excepte isomorfisme, els semigrups numèrics classifiquen el conjunt de submonoides de $\mathbb{N}$.

A partir dels resultats següents estudiem en més profunditat els generadors dels monoides i dels semigrups numèrics denotant $S^*=S\setminus\{0\}$ i $S^*+S^*=\{a+b\mid s_1,s_2\in S^*\}$.

Lema 5.1.4 ([6], Lemma 2.3)**.** *Siga S un submonoide de $\mathbb{N}$. Aleshores $S^*\setminus(S^*+S^*)$ és un sistema de generadors de S. És més, cada sistema de generadors de S conté $S^*\setminus(S^*+S^*)$.*

Demostració. Siga $s\in S^*$. Si $s\notin S^*\setminus(S^*+S^*)$, aleshores $s\in S^*+S^*$ i per tant $s=a+b$ amb $a,b\in S^*$. Com $a,b<s$, podem repetir aquest procediment fins que no siga possible després d'un nombre finit de passos. En aquest punt, tindrem $s_1,\dots,s_n\in S^*\setminus(S^*+S^*)$ tals que $s=s_1+\cdots+s_n$. Aquest fet prova que $S^*\setminus(S^*+S^*)$ és un sistema de generadors. $\square$

Introduirem ara una ferramenta essencial en els semigrups numèrics i que permetrà demostrar el resultat fonamental respecte de la generació dels semigrups numèrics. Siga S un semigrup numèric i siga $n \in S^*$. El conjunt d'Apéry de n en S és

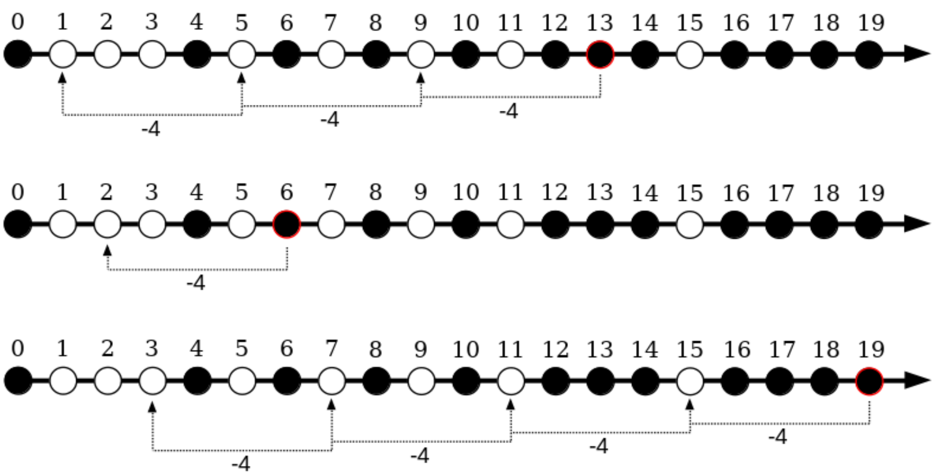
$$\mathrm{Ap}(S, n) = \{s \in S \mid s - n \notin S\}.$$

Lema 5.1.5 ([6], Lemma 2.4)**.** *Siga S un semigrup numèric i $n \in S^*$. Aleshores, per a tota $i \in \{1, \ldots, n-1\}$, $\mathrm{Ap}(S, n) = \{0 = w(0), w(1), \ldots, w(n-1)\}$ on $w(i)$ és el menor element de S congruent amb i módul n.*

Demostració. Siga $a \in \mathrm{Ap}(S, n)$. Per l'algoritme de divisió, tenim que $a = qn + r$ amb $q, r \in \mathbb{N}$ i $0 \leq r < n$. Com $a \in \mathrm{Ap}(S, n)$, efectivament $a - n \notin S$, és a dir, $a - n = (q-1)n + r \notin S$. Així doncs, $a = qn + r$ és el menor element de S congruent amb r módul n per a $r \in \{1, \ldots, n\}$.

Siga $w(i)$ el menor element de S congruent amb i módul n per a $i \in \{1, \ldots, n\}$. Aleshores, $w(i) - n \notin S$ per la minimalitat de $w(i)$ i per tant $w(i) \in \mathrm{Ap}(S, n)$. □

Exemple 5.1.6. Siga S el semigrup numèric generat per $\{4, 6, 13\}$. Aleshores, podem escriure $S = \{0, 4, 6, 8, 10, 12, 13, 14, 16, \ldots\}$. Per tant, $\mathrm{Ap}(S, 4) = \{w(0) = 0, w(1) = 13, w(2) = 6, w(3) = 19\}$.



El Lema 5.1.5 implica trivialment que la cadinalitat de $\mathrm{Ap}(S, n)$ és n. A partir d'aquest fet, obtenim el següent resultat.

Lema 5.1.7 ([6], Lemma 2.6)**.** *Siga S un semigrup numèric i siga $n \in S^*$. Aleshores, per a tota $s \in S^*$, existeix un únic $(k, w) \in \mathbb{N} \times \mathrm{Ap}(S, n)$ tal que $s = kn + w$.*

Demostració. Per a tot $s \in S^*$, tenim que s és congruent amb i mòdul n per a $i \in \{1, \ldots, n\}$. Si $s \in \mathrm{Ap}(S, n)$, aleshores s serà el menor element que ho complisca. Si $s \notin \mathrm{Ap}(S, n)$, per la minimalitat de w sabem que $w < s$ i tindrem que $s = kn + w$ amb $k \in \mathbb{N} \setminus \{0\}$. □

Siga S un semigrup numèric. Diguem que un sistema de generadors de S és *minimal* si cap dels seus subconjunts genera a S.

Teorema 5.1.8 ([6], Theorem 2.7). *Cada semigrup numèric admet un únic sistema minimal de generadors. Aquest sistema minimal de generadors és finit.*

Demostració. Pel Lema 5.1.4 sabem que $S^* \setminus (S^* + S^*)$ és el sistema minimal de generadors de S perquè tot sistema de generadors de S l'inclou. Del Lema 5.1.7 deduïm que per a qualsevol $s \in S^*$, tenim que $S = \langle \mathrm{Ap}(S, s) \cup \{s\} \rangle$. Com $\mathrm{Ap}(S, s) \cup \{s\}$ és finit, tenim que $S^* \setminus (S^* + S^*)$ també és finit. □

Siga S un semigrup numèric i siga $\{a_1, a_2, \ldots, a_n\}$ el seu sistema minimal de generadors. Aleshores, anomenem a_1 com la *multiplicitat* de S, denotada com $\mathrm{m}(S)$. El nombre n és la cardinalitat del sistema de generadors minimal al qual anomenem *dimensió d'escabussament* i denotem per $\mathrm{e}(S)$.

Proposició 5.1.9 ([6], Proposition 2.10). *Siga S un semigrup numèric. Aleshores*

1) $\mathrm{m}(S) = \text{mín } (S \setminus \{0\})$,

2) $\mathrm{e}(S) \leq \mathrm{m}(S)$.

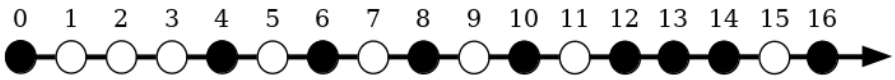
Demostració. Com $\mathrm{m}(S) \in S^*$, tenim que $\mathrm{m}(S) \geq \text{mín } S^*$. Però, si $\mathrm{m}(S) > \text{mín } S^*$, aleshores mín S^* seria un generador minimal menor que $\mathrm{m}(S)$. Per definició de multiplicitat, açò seria una contradicció i conseqüentment $\mathrm{m}(S) = \text{mín } S^*$.

Pel Lema 5.1.7, deuïm que $\mathrm{Ap}\big(S, \mathrm{m}(S)\big) \cup \{\mathrm{m}(S)\}$ és un sistema de generadors de S amb cardinalitat $\mathrm{m}(S)$. Aleshores, per la definició de sistema minimal de generadors, tenim que $\mathrm{e}(S) \leq \mathrm{m}(S)$. □

5.2 Invariants principals

Sent S un semigrup numèric, el conjunt d'elements $\mathrm{C}(S) = \mathbb{N} \setminus S$ és conegut com el conjunt de *clots* de S i la seua cardinalitat és el *gènere* de S que denotarem com $\mathrm{g}(S)$. Com que és un conjunt finit, per la definició de semigrup numèric, el seu major element és anomenat el nombre de Frobenius del semigrup S i denotat com $\mathrm{F}(S)$. En la literatura, el concepte de nombre de Frobenius és substituït de vegades pel concepte de *conductor* de S, denotat per $\mathrm{c}(S)$, que es defineix com el menor element $x \in S$ tal que $x+n \in S$ per a tota $n \in \mathbb{N}$. Note's que el conductor d'un semigrup numèric és $\mathrm{F}(S)+1$.

Exemple 5.2.1. Siga S el semigrup numèric generat per $\{4, 6, 13\}$. Recordem que podem escriure $S = \{0, 4, 6, 8, 10, 12, 13, 14, 16, \dots\}$. Aleshores, $\mathrm{C}(S) = \{1, 2, 3, 5, 7, 9, 11, 15\}$, $\mathrm{F}(S) = 15$ i $\mathrm{g}(S) = 8$.



No existeixen fórmules generals per a calcular el nombre de Frobenius o el gènere per a semigrups numèrics amb dimensió d'escabussament superior a dos però, si es coneix el conjunt d'Apéry per a un element del semigrup numèric, es poden calcular les dues invariants gràcies al resultat següent.

Proposició 5.2.2 ([6], Proposition 2.12)**.** *Siga S un semigrup numèric i siga $n \in S^*$. Aleshores*

1) $\mathrm{F}(S) = (\text{màx}\ \mathrm{Ap}(S, n)) - n$,

2) $\mathrm{g}(S) = \frac{1}{n}\left(\sum_{w \in \mathrm{Ap}(S,n)} w\right) - \frac{n-1}{2}$.

Demostració. Siga $x \in \mathbb{N}$, suposem $x > (\text{màx}\ \mathrm{Ap}(S, n)) - n$. Per tant, $x + n > (\text{màx}\ \mathrm{Ap}(S, n))$. Siga $w \in \mathrm{Ap}(S, n)$ congruent amb $x+n$ módul n. Com $w < x+n$, aleshores $x-n = w+kn$ amb $k \in \mathbb{N} \setminus \{0\}$. Així doncs, $x = w + (k-1)n \in S$. És a dir, que qualsevol $x > (\text{màx}\ \mathrm{Ap}(S, n)) - n$ pertany a S, fent que $(\text{màx}\ \mathrm{Ap}(S, n)) - n$ siga per definició el nombre de Frobenius.

Per a cada $w \in \mathrm{Ap}(S, n)$, tenim que w és congruent amb i módul n per a $i \in \{0, 1, \dots, n-1\}$. És a dir, que $w = i + kn$ amb $k \in \mathbb{N} \setminus \{0\}$. Així doncs, podem escriure

$$\mathrm{Ap}(S, n) = \{w(0) = 0, w(1) = k_1 n + 1, \dots, w(n-1) = k_{n-1} n + n - 1\}.$$

Com $w(i)$ és el menor element de S congruent amb i módul n, $w(i)$ és el representant de la i-ena classe módul n amb k_i clots en ella. Per tant, per definició de gènere de S i sabent que

$\sum_{j=1}^{n} j = \frac{n(n+1)}{2}$, tenim que

$$\begin{aligned}
\mathrm{g}(S) &= k_1 + \cdots + k_{n-1} = \frac{1}{n}(k_1 n + \cdots + k_{n-1} n) \\
&= \frac{1}{n}(k_1 n + \cdots + k_{n-1} n) + \frac{1}{n}\sum_{j=1}^{n-1} j - \frac{1}{n}\sum_{j=1}^{n-1} j \\
&= \frac{1}{n}(k_1 n + \cdots + k_{n-1} n) + \frac{1}{n}(1 + \cdots + n - 1) - \frac{1}{n}\frac{(n-1)n}{2} \\
&= \frac{1}{n}(k_1 n + 1 + \cdots + k_{n-1} n + n - 1) - \frac{n-1}{2} = \frac{1}{n}\sum_{w \in \mathrm{Ap}(S,n)} w - \frac{n-1}{2}.
\end{aligned}$$

□

En el cas particular dels semigrups numèrics amb dimensió d'escabussament dos, sí que es coneix el conjunt d'Apéry en funció de la multiplicitat.

Lema 5.2.3 ([7], Theorem 1)**.** *Siga $S = \langle \alpha, \beta \rangle$, és a dir, el semigrup numèric S està generat minimalment per α i β. Aleshores,* $\mathrm{Ap}(S, \alpha) = \{0, \beta, \ldots, (\alpha - 1)\beta\}$.

Demostració. Primer vegem que $\alpha \notin \mathrm{Ap}(S, \alpha)$ perquè $\alpha - \alpha = 0 \in S$. Ara provem que $\beta \in \mathrm{Ap}(S, \alpha)$. Si $\beta \notin \mathrm{Ap}(S, \alpha)$, aleshores $\beta - \alpha \in S$. Açò implica que $\beta - \alpha = \lambda\alpha - \mu\beta$ amb $\lambda, \mu \in \mathbb{N}$. És a dir, que $\beta(1 - \mu) = \alpha(\lambda + 1) \in \mathbb{N}$ i per tant, $\mu = 0$ o $\mu = 1$.

Si $\mu = 0$, tenim que $\beta = \alpha(\lambda + 1)$ i així $\alpha \mid \beta$ que és una contradicció perquè $\mathrm{mcd}(\alpha, \beta) = 1$ al ser generadors minimals de S. Si $\mu = 1$, tenim que $0 = \alpha(\lambda + 1) \Leftrightarrow -\alpha = \alpha\lambda \in \mathbb{N}$ que també és una contradicció. Per tant, $\beta \in \mathrm{Ap}(S, \alpha)$. Així doncs, deduïm que $k\beta \in \mathrm{Ap}(S, \alpha)$ amb $k = \{0, \ldots, \alpha - 1\}$ perquè $\alpha\beta \in \mathrm{Ap}(S, \alpha)$, ja que $\alpha\beta - \alpha = \alpha(\beta - 1) \in S$. Com la cardinalitat de $\mathrm{Ap}(S, \alpha)$ és α, tenim que $\mathrm{Ap}(S, \alpha) = \{0, \beta, \ldots, (\alpha - 1)\beta\}$. □

Aquest resultat permet calcular el nombre de Frobenius i el gènere en semigrups numèrics amb dimensió d'escabussament dos.

Corol·lari 5.2.4. *Siga $S = \langle \alpha, \beta \rangle$. Aleshores*

1) $\mathrm{F}(\langle \alpha, \beta \rangle) = \alpha\beta - \alpha - \beta$,

2) $\mathrm{g}(\langle \alpha, \beta \rangle) = \dfrac{\alpha\beta - \alpha - \beta + 1}{2}$.

Demostració. Com $\mathrm{Ap}(S, \alpha) = \{0, \beta, \ldots, (\alpha - 1)\beta\}$, pel Lema 5.2.3, tenim que

$$\mathrm{F}(\langle \alpha, \beta \rangle) = (\text{màx } \mathrm{Ap}(S, \alpha)) - \alpha = (\alpha - 1)\beta - \alpha = \alpha\beta - \alpha - \beta.$$

Tanmateix, també tenim que

$$\begin{aligned}
\mathrm{g}\big(\langle \alpha, \beta\rangle\big) &= \frac{1}{\alpha} \sum_{w \in \mathrm{Ap}(S,n)} w - \frac{\alpha-1}{2} \\
&= \frac{\beta}{\alpha}\left(\frac{(\alpha-1)\alpha}{2}\right) - \frac{\alpha-1}{2} \\
&= \frac{\beta(\alpha-1)-\alpha+1}{2} = \frac{\alpha\beta-\alpha-\beta+1}{2}.
\end{aligned}$$

□

Note's que en el cas de semigrups numèrics amb $\mathrm{e}(S) = 2$, tenim que $\mathrm{g}(S) = (\mathrm{F}(S) + 1)/2$. Aquest resultat no és general per a semigrups numèrics amb dimensió d'escabussament superior però sí que és certa la desigualtat com veurem més endavant.

Lema 5.2.5. *Siga S un semigrup numèric. Si $s \in S$, aleshores $F(S) - s \notin S$.*

Demostració. Suposem que $\mathrm{F}(S) - s \in S$. Per la Proposició 5.2.2, tenim que

$$\mathrm{F}(S) - s \in S \Leftrightarrow (\text{màx } \mathrm{Ap}(S, n)) - n - s \in S \Leftrightarrow \big(\text{màx } \mathrm{Ap}(S, n)\big) - (n + s) \in S.$$

Com $n \in S^*$ i $s \in S$, resulta que $n + s \in S$. Per tant, per la definició de conjunt d'Apéry, obtenim una contradicció perquè $\big(\text{màx } \mathrm{Ap}(S, n)\big) - (n + s) \notin S$. □

A partir d'aquest resultat obtenim la relació entre el gènere i el nombre de Frobenius per a qualsevol semigrup numèric.

Teorema 5.2.6 ([6], Lemma 2.14)**.** *Siga S un semigrup numèric. Aleshores*

$$\mathrm{g}(S) \geq \frac{F(S) + 1}{2}.$$

Demostració. Siga $s \in S^*$ amb $s < \mathrm{F}(S)$. Pel Lema 5.2.5 sabem que $\mathrm{F}(S) - s \notin S$ i també tenim que $\mathrm{F}(S) - s < \mathrm{F}(S)$. Per tant, cada s es relaciona amb un clot de s fent que la cardinalitat de $\mathrm{C}(S)$ siga com a màxim $\frac{\mathrm{F}(S)+1}{2}$. És a dir, que $\mathrm{e}(S) \leq \frac{\mathrm{F}(S)+1}{2}$. □

Capítol 6

Semimóduls i sizígies en semigrups numèrics amb dos generadors minimals

En aquest capítol estudiarem en profunditat l'estructura de semimódul o ideal relatiu en semigrups numèrics. També veurem la relació entre un semimódul i la seua sizígia associada per mitjà dels parells fonamentals. Tot açò en el context dels semigrups numèrics amb dimensió d'escabussament dos dels que presentarem algun resultat particular. Utilitzarem les seccions 3 i 4 de [8] com a referència principal així com [9] i [10] quan parlem dels clots de semigrups numèrics generats minimalment per dos elements.

6.1 Generació de semimóduls

Siga S un semigrup numèric. Un *semimódul* Δ de S és un subconjunt no buit de $\mathbb{N}$ tal que $\Delta + S \subseteq \Delta$. És a dir, que tot element de Δ sumat a qualsevol element de S, ha de pertànyer a Δ. Un *sistema de generadors* de Δ és un subconjunt $\mathcal{E} \subseteq \Delta$ que compleix

$$\bigcup_{x \in \mathcal{E}} (x + S) = \Delta.$$

Just com passava amb els generadors d'un semigrup numèric, un sistema de generadors $\mathcal{E}$ d'un semimódul Δ es diu *minimal* si cap subconjunt de $\mathcal{E}$ genera a Δ. Note's que com $\Delta \setminus S$ és finit, cada semimódul de S està finitament generat.

Proposició 6.1.1 ([8], Lemma 2.1)**.** *Cada semimódul Δ d'un semigrup numèric S admet un únic sistema minimal de generadors.*

Demostració. L'Algoritme 1 mostra el procediment d'obtenció de l'únic sistema minimal de generadors per a un semimódul Δ. Note's que el sistema de generadors és minimal perquè es van afegint element d'un en un al conjunt de generadors i immediatament després es comprova que aquest conjunt genere a Δ. A més, el sistema és únic perquè a cada iteració només hi ha un possible element a afegir al conjunt de generadors. Finalment, com tot semimódul està finitament generat, s'arribarà a un sistema de generadors que genere tot Δ en un nombre finit de passos. $\square$

Algoritme 1: Obtenció del sistema minimal de generadors d'un semimódul

---

Entrada: Semimódul Δ d'un semigrup numèric S
Eixida : El sistema de generadors minimal $\mathcal{E} = \{x_1, \ldots, x_n\}$ del semimódul Δ
$\mathcal{E} := \{x_1 := \min(\Delta)\}$;
Mentre $\bigcup_{x\in\mathcal{E}}(x + S) \neq \Delta$ **fer**
 $\mathcal{E} = \mathcal{E} \cup \min\big(\Delta \setminus \bigcup_{x\in\mathcal{E}}(x + S)\big)$;

D'aquesta manera podem identificar cada semimódul amb el seu sistema minimal de generadors per facilitar l'operativa en ells. A més, aquest sistema compta amb una propietat molt interessant que facilitarà el càlcul computacional de semimóduls en un semigrup numèric.

Proposició 6.1.2 ([8], Lemma 2.2)**.** *Siga S un semigrup numèric i Δ un semigrup numèric de S. Siga $\{x_1, \ldots, x_n\}$ el sistema minimal de generadors de Δ. Aleshores, $|x_i - x_j| \in C(S)$ per a tota $i \neq j$. Recíprocament, qualsevol subconjunt $\{x_1, \ldots, x_n\} \subseteq \mathbb{N}$ que complisca aquesta propietat, genera minimalment a Δ.*

Demostració. Podem suposar sense pèrdua de generalitat que $x_j > x_i$ llavors $|x_i - x_j| = x_i - x_j$. Si $\{x_1, \ldots, x_n\}$ són generadors minimals de Δ, suposem que $x_i - x_j \in S$. Per tant, $x_i \in x_j + S$ i podríem generar x_i a partir de x_j. No obstant, aquest fet és una contradicció perquè x_i és un generador minimal. Per tant, $x_i - x_j \notin S$ i com $x_i - x_j > 0$, tenim que $x_i - x_j \in \mathrm{C}(S)$.

Si $x_i - x_j \in \mathrm{C}(S)$, donat $\mathcal{E} = \{x_1, \ldots, x_n\}$, sempre podem formar el conjunt

$$\Delta = \bigcup_{x\in\mathcal{E}}(x + S)$$

que és el semimódul Δ generat per $\mathcal{E}$. Per demostrar la minimalitat del sistema de generadors $\mathcal{E}$, demostrarem que $x_i \notin \bigcup_{j\neq i}(x_j + S)$. És a dir, que cap x_i es pot generar a partir d'altre element de $\mathcal{E}$. Suposem $x_i \in \bigcup_{j\neq i}(x_j + S)$. Per tant, existeix $x_j \in \mathcal{E}$ tal que $x_i = x_j + s$ amb $s \in S$. Açò implica que $x_i - x_j = s$ però $x_i - x_j \in \mathrm{C}(S)$ i per tant arribem a una contradicció. Per tant, $x_i \notin \bigcup_{j\neq i}(x_j + S)$ i $\mathcal{E}$ és un sistema de generadors minimal del semimódul Δ. $\square$

Dos semimóduls Δ i Δ' d'un semigrup S es diuen *isomòrfics* si existeix $n \in \mathbb{Z}$ tal que l'aplicació $f\colon x \mapsto x + n$ entre Δ i Δ' és bijectiva. Per a cada semimódul Δ de S existeix un únic semimódul isomòrfic que conté a 0. Aquest semimódul s'anomena *normalitzat* i és de la forma $\Delta^\circ \coloneqq \{x - \text{mín}\, \Delta^\circ \mid x \in \Delta\}$.

6.2 Sizígies i parells fonamentals

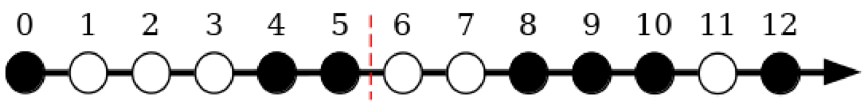
Siga $S = \langle \alpha, \beta \rangle$ un semigrup numèric i Δ un semimódul de S amb sistema minimal de generadors $I = \{x_1, \ldots, x_n\}$. La *sizígia* de Δ és el semimódul

$$\operatorname{Siz}(\Delta) \coloneqq \bigcup_{x_i, x_j \in I, i \neq j} \Big(\big(x_i + \langle \alpha, \beta \rangle\big) \cap \big(x_j + \langle \alpha, \beta \rangle\big) \Big).$$

És a dir, el semimódul $\operatorname{Siz}(\Delta)$ està format pels elements de Δ que admeten més d'una representació de la forma $x + s$ amb $x \in I$ i $s \in S$.

Per demostrar el resultat principal de la secció, hem de presentar una família de semigrups numèrics molt estudiada en el context dels semigrups numèrics. Siga S un semigrup numèric, diguem que S és *simètric* si per a tot $x \in C(S)$, tenim que $\mathrm{F}(S) - x \in S$.

Exemple 6.2.1. Siga $S = \langle 4, 5 \rangle$ un semigrup simètric. Tenim que $\mathrm{C}(S) = \{1, 2, 3, 6, 7, 11\}$ i $\mathrm{F}(S) = 11$.



$$\begin{array}{ll} 11 - 11 = 0 \in S & 11 - 3 = 8 \in S \\ 11 - 7 = 4 \in S & 11 - 2 = 9 \in S \\ 11 - 6 = 5 \in S & 11 - 1 = 10 \in S \end{array}$$

Lema 6.2.2. *Tot semigrup numèric amb dimensió d'escabussament dos és simètric.*

Demostració. Siga $S = \langle \alpha, \beta \rangle$, pel Corol·lari 5.2.4 sabem que $\mathrm{g}(S) = \frac{\mathrm{F}(S)+1}{2}$. Açò implica que la cardinalitat del conjunt $\{s \in S \mid s < \mathrm{F}(S)\}$ és $\mathrm{g}(S)$. És a dir, que la meitat d'elements de $\mathbb{N}$ menors que el conductor de S són clots i l'altra meitat són elements de S. Del Lema 5.2.5, deduïm que per a tot $s \in S$ amb $s < \mathrm{F}(S)$, tenim que $\mathrm{F}(S) - s \notin S$. Per tant, tots els clots de S són de la forma $\mathrm{F}(S) - s$ per a algun $s < \mathrm{F}(S)$. Així doncs, per a tot $\mathrm{F}(S) - s \in \mathrm{C}(S)$, tenim que $\mathrm{F}(S) - \big(\mathrm{F}(S) - s\big) = s \in S$ i S és simètric. $\square$

Els semigrups numèrics amb dimensió d'escabussament dos han sigut àmpliament estudiats per tractar-se de semigrups numèrics relativament senzills d'analitzar. Tant és així que fins i tot els clots d'aquests semigrups numèrics es poden caracteritzar segons la següent propietat.

Lema 6.2.3 ([9], Lemma 1)**.** *Siga $S = \langle \alpha, \beta \rangle$ un semigrup numèric i $x \in \mathbb{N}$. Aleshores, $x \in \mathbb{Z} \setminus S$ si i sols si existeixen $a, b \in \mathbb{N} \setminus \{0\}$ tals que $x = \alpha\beta - a\alpha - b\beta$.*

Demostració. Suposem $x \in \mathbb{Z} \setminus S$. Pel Lema 6.2.2, tenim que S és simètric i pel Corol·lari 5.2.4 sabem que $F(S) = \alpha\beta - \alpha - \beta$. Per tant, $\alpha\beta - \alpha - \beta - x \in S$. Açò implica que existeixen $\lambda, \mu \in \mathbb{N}$ tals que $\alpha\beta - \alpha - \beta - x = \lambda\alpha + \mu\beta$. Aïllem x i obtenim que $x = \alpha\beta - (\lambda+1)\alpha - (\mu+1)\beta$. Per tant, si denotem $a \coloneqq \lambda + 1$ i $b \coloneqq \mu + 1$, tenim que $x = \alpha\beta - a\alpha - b\beta$.

Suposem que $x = \alpha\beta - a\alpha - b\beta$ per alguns $a, b \in \mathbb{N} \setminus \{0\}$. Si $x \in S$, aleshores existeixen $\lambda, \mu \in \mathbb{N}$ tals que $x = \alpha\beta - a\alpha - b\beta = \lambda\alpha + \mu\beta$. Operant un poc obtenim que

$$\begin{aligned}\alpha\beta - a\alpha - b\beta = \lambda\alpha + \mu\beta &\Leftrightarrow \alpha\beta - (a-1)\alpha - \alpha - (b-1)\beta - \beta = \lambda\alpha + \mu\beta \\ &\Leftrightarrow \alpha\beta - \alpha - \beta = (\lambda + a - 1)\alpha + (\mu + b - 1)\beta.\end{aligned}$$

Com $(\lambda + a - 1), (\mu + b - 1) > 0$, tenim que $= \alpha\beta - \alpha - \beta \in S$ però açò és una contradicció perquè $\mathrm{F}(S) = \alpha\beta - \alpha - \beta$. Per tant, $x \notin S$. $\square$

A més a més, el resultat anterior ens permet obtindre una manera fàcil de comprovar la propietat descrita en la Proposició 6.1.2.

Lema 6.2.4 ([10], Lemma 3.19)**.** *Siga $S = \langle \alpha, \beta \rangle$ un semigrup numèric. Siguen $x_1 = \alpha\beta - a_1\alpha - b_1\beta$ i $x_2 = \alpha\beta - a_2\alpha - b_2\beta$ clots de S. Aleshores, $|x_1 - x_2| \in C(S)$ si i sols si $(a_2 - a_1)(b_2 - b_1) < 0$.*

Demostració. Podem suposar sense pèrdua de generalitat que $|x_1 - x_2| = x_1 - x_2$. Suposem $x_1 - x_2 \in \mathrm{C}(S)$. Com $x_1, x_2 \in \mathrm{C}(S)$, tenim que

$$x_1 - x_2 = \alpha\beta - a_1\alpha - b_1\beta - \alpha\beta + a_2\alpha + b_2\beta = (a_2 - a_1)\alpha + (b_2 - b_1)\beta.$$

Si $(a_2 - a_1) < 0$ i $(b_2 - b_1) < 0$, aleshores $x_1 - x_2 < 0$ i $x_1 - x_2 \notin \mathrm{C}(S)$ però sabem que $x_1 - x_2 \in \mathrm{C}(S)$. Si $(a_2 - a_1) > 0$ i $(b_2 - b_1) > 0$, aleshores $x_1 - x_2 \in S$ però sabem que $x_1 - x_2 \notin S$ perquè $x_1 - x_2 \in \mathrm{C}(S)$. Per tant, al descartar aquestes dues situacions per conduir a contradiccions, només queden dues possibilitats pel que fa al signe de $(a_2 - a_1)$ i $(b_2 - b_1)$ que es resumeixen en $(a_2 - a_1)(b_2 - b_1) < 0$.

Primer vegem que $a_1, a_2 < \beta$ i que $b_1, b_2 < \alpha$. Podem escriure

$$x_1 = \alpha\beta - a_1\alpha - b_1\beta = \alpha(\beta - a_1) - b_1\beta \in \mathrm{C}(S).$$

Si $a_1 \geq \beta$, $x_1 < 0$ i $x_1 \notin \mathrm{C}(S)$. Per tant, $a_1 < \beta$. També podem escriure

$$x_1 = \alpha\beta - a_1\alpha - b_1\beta = -a_1\alpha + \beta(\alpha - b_1) \in \mathrm{C}(S).$$

Si $b_1 \geq \alpha$, $x_1 < 0$ i $x_1 \notin \mathrm{C}(S)$. Per tant, $b_1 < \alpha$. Anàlogament, demostraríem $a_2 < \beta$ i $b_2 < \alpha$. Si denotem $a \coloneqq a_2 - a_1$ i $b \coloneqq b_2 - b_1$, podem escriure

$$x_1 - x_2 = a\alpha - b\beta = \alpha\beta - \alpha\beta + a\alpha + b\beta = \alpha\beta - (\beta - a)\alpha + b\beta.$$

Si $b < 0$ i $a > 0$, com $(\beta - a) > 0$ perquè $a_2 < \beta$ i $1 \leq a_1 < \beta$, tenim pel Lema 6.2.3 que $x_1 - x_2 \in \mathrm{C}(S)$. També podem escriure

$$x_1 - x_2 = a\alpha - b\beta = \alpha\beta - \alpha\beta + a\alpha + b\beta = \alpha\beta + a\alpha - (\alpha - b)\beta.$$

Si $b > 0$ i $a < 0$, com $(\alpha - b) > 0$ perquè $b_2 < \alpha$ i $1 \leq b_1 < \alpha$, també tenim que $x_1 - x_2 \in \mathrm{C}(S)$. Per tant, si $(a_2 - a_1)(b_2 - b_1) < 0$, $x_1 - x_2 \in \mathrm{C}(S)$. □

Siga S un semigrup numèric. Un *parell fonamental* $[I, J]$ de S consisteix en dues seqüències $I = (i_k)_{k=0}^n$ i $J = (j_k)_{k=0}^n$ tals que

1) $i_0 = 0$,

2) $i_1, \ldots, i_n, j_1, \ldots, j_{n-1} \in C(S)$ i $j_0, j_n \leq \alpha\beta$,

3)
$$\begin{array}{llll} i_k \equiv j_k & \text{mód } \alpha & \text{amb} \quad i_k < j_k & \text{per a } k = 0, \ldots, n; \\ j_k \equiv i_{k+1} & \text{mód } \beta & \text{amb} \quad j_k > i_{k+1} & \text{per a } k = 0, \ldots, n-1; \\ j_n \equiv i_0 & \text{mód } \beta & \text{amb} \quad j_n \geq i_0, & \end{array}$$

4) $|i_k - i_l| \in C(S)$ per a $1 \leq k < l \leq n$.

En el cas dels semigrups numèrics amb dimensió d'escabussament dos, els elements de J es poden escriure en funció dels dos generadors minimals i de la representació dels elements de I com a clots d'acord amb el Lema 6.2.3.

Lema 6.2.5 ([8], Lemma 4.1)**.** *Siga* $[I, J]$ *un parell fonamental de* $S = \langle \alpha, \beta \rangle$ *amb* $I = \{i_0 = 0, \ldots, i_n\}$ *i* $J = \{j_0, \ldots, j_n\}$*. Aleshores*

$$\begin{aligned} j_0 &= \alpha\beta - a_1\alpha, \\ j_k &= \alpha\beta - a_{k+1}\alpha - b_k\beta \quad \textit{per a } k = 1 \ldots, n-1; \\ j_n &= \alpha\beta - b_n\beta. \end{aligned}$$

Demostració. Per definició de parell fonamental, tenim que $j_0 \equiv 0 \text{ mód } \alpha$ amb $j_0 > 0$, és a dir, que $j_0 = \lambda_0\alpha$ amb $0 \leq \lambda_0 \leq \beta$. Però també tenim que $j_0 \equiv i_1 \text{ mód } \beta$ amb $j_0 > i_1$ o equivalentment $j_0 = i_1 + \mu_0\beta$ amb $0 \leq \mu_0 < \alpha$. Com $i_1 \in \mathrm{C}(S)$, pel Lema 6.2.3, tenim que

$$j_0 = i_1 + \mu_0\beta = \alpha\beta - a_1\alpha - b_1\beta + \mu_0\beta = (\beta - a_1)\alpha + (\mu_0 - b_1)\beta = \lambda_0\alpha.$$

Per tant, deduïm que $j_0 = (\beta - a_1)\alpha = \alpha\beta - a_1\alpha$.

Repetint el raonament anterior per a j_n, tenim que $j_n = i_n$ mód $\alpha = i_0$ mód β i per tant

$$j_n = i_n + \lambda_n \alpha = \alpha\beta - a_n\alpha - b_n\beta + \lambda_n\alpha = (\lambda_n - a_n)\alpha + (\alpha - b_n)\beta = \mu_n\beta.$$

Així doncs, deduïm que $j_n = (\alpha - b_n)\beta = \alpha\beta - b_n\beta$.

Per definició de parell fonamental, tenim que $j_k \in \mathrm{C}(S)$ per a $k = 1, \ldots, n-1$ i pel Lema 6.2.3, podem escriure $j_k = \alpha\beta - a_{j_k}\alpha - b_{j_k}\beta$ amb $a_{j_k}, b_{j_k} \in \mathbb{N} \setminus \{0\}$. Com $j_k \equiv i_k$ mód α i $j_k > i_k$, aleshores, sent $\lambda_k \in \mathbb{N}$, tenim que

$$\begin{aligned} j_k = i_k + \lambda_k\alpha \Leftrightarrow j_k - i_k &= \alpha\beta - a_{j_k}\alpha - b_{j_k}\beta - \alpha\beta + a_k\alpha - b_k\beta \\ &= (a_k - a_{j_k})\alpha + (b_k - b_{j_k})\beta = \lambda_k\alpha. \end{aligned}$$

Per tant, deduïm que $b_{j_k} = b_k$ i treballant de manera anàloga a partir de $j_k \equiv i_{k+1}$ mód β amb $j_k > i_{k+1}$, tindriem que $a_{j_{k+1}} = a_{k+1}$. És a dir, que $j_k = \alpha\beta - a_{j_k}\alpha - b_{j_k}\beta = \alpha\beta - a_{k+1}\alpha - b_k\beta$. $\square$

L'escriptura dels elements de J de la forma presentada en el lema anterior serà una ferramenta molt útil per demostrar el resultat essencial de la secció.

Teorema 6.2.6 ([8], Theorem 4.3)**.** *Siga $S = \langle \alpha, \beta \rangle$ un semigrup numèric. Siga $[I, J]$ un parell fonamental de S i Δ el semimódul de S generat pels elements de I. Aleshores*

$$\mathrm{Siz}(\Delta) = \bigcup_{0 \le k < m \le n} \Big(\big(i_k + \langle \alpha, \beta \rangle\big) \cap \big(i_m + \langle \alpha, \beta \rangle\big) \Big) = \bigcup_{k=0}^{n} \big(j_k + \langle \alpha, \beta \rangle\big).$$

Demostració. Primer considerem l'element $j_k + \langle \alpha, \beta \rangle$ per demostrar la inclusió $\supseteq$. Per definició de parell fonamental, tenim que $j_k \equiv i_k$ mód α i $j_k \equiv i_{k+1}$ mód β. És a dir, que $j_k = i_k + \lambda\alpha = i_{k+1} + \mu\beta$ amb $\lambda, \mu \in \mathbb{N}$. Per tant,

$$j_k + \langle \alpha, \beta \rangle \in \bigcup_{0 \le k < m \le n} \Big(\big(i_k + \langle \alpha, \beta \rangle\big) \cap \big(i_m + \langle \alpha, \beta \rangle\big) \Big).$$

Considerem ara l'element $i_m + \gamma_m \in i_m + \langle \alpha, \beta \rangle$. Per a cada $\gamma_m \in \langle \alpha, \beta \rangle$, tenim que

$$i_m + \gamma_m \in \big(i_k + \langle \alpha, \beta \rangle\big) \cap \big(i_m + \langle \alpha, \beta \rangle\big) \Leftrightarrow i_m + \gamma_m \in i_k + \langle \alpha, \beta \rangle \Leftrightarrow \gamma_m \in \langle \alpha, \beta \rangle + i_k - i_m.$$

Per definició de parell fonamental, $i_m, i_k \in \mathrm{C}(S)$ i, pel Lema 6.2.3, sabem que

$$\begin{aligned} i_k &= \alpha\beta - a_k\alpha - b_k\beta, \\ i_m &= \alpha\beta - a_m\alpha - b_m\beta \end{aligned}$$

i per tant

$$\begin{aligned}i_m - i_k &= \alpha\beta - a_m\alpha - b_m\beta - \alpha\beta + a_k\alpha + b_k\beta = (a_k - a_m)\alpha + (b_k - b_m)\beta \\ &= (a_k - a_m)\alpha + (b_k - b_m)\beta + \alpha\beta - \alpha\beta = \alpha\beta - (\beta - a_k + a_m)\alpha - (b_m - b_k)\beta.\end{aligned}$$

Com $i_m - i_k \in \mathrm{C}(S)$, pel Lema 6.2.3 sabem que $\beta - a_k + a_m > 0$ i $b_m - b_k > 0$. Pel Lema 6.2.4, podem suposar $a_m < a_k$ i $b_m > b_k$ i tenim que existeixen $\lambda, \mu \in \mathbb{N}$ tals que

$$\gamma_m = \lambda\alpha + \mu\beta + i_k - i_m = \lambda\alpha + \mu\beta - \alpha\beta + (\beta - a_k + a_m)\alpha + (b_m - b_k)\beta.$$

D'una banda, traient factor común α, tenim que

$$\gamma_m = (\beta - a_k + a_m + \lambda - \beta)\alpha + (\mu + b_m - b_k)\beta = (\lambda - a_k + a_m)\alpha + (\mu + b_m - b_k)\beta.$$

D'altra banda, traient factor común β, tenim que

$$\gamma_m = (\beta - a_k + a_m + \lambda)\alpha + (\mu + b_m - b_k - \alpha)\beta.$$

Com $\gamma_m \in \langle \alpha, \beta \rangle$, aleshores deduïm que

$$\begin{aligned}\lambda - a_k + a_m &\geq 0, \\ \mu + b_m - b_k &\geq 0, \\ \beta - a_k + a_m + \lambda &\geq 0, \\ \mu + b_m - b_k - \alpha &\geq 0.\end{aligned}$$

Podem reescriure γ_m com

$$\begin{aligned}\gamma_m &= \lambda\alpha + \mu\beta - \alpha\beta + (\beta - a_k + a_m)\alpha + (b_m - b_k)\beta \\ &= (\lambda - a_k + a_m)\alpha + \mu\beta + (b_m - b_k)\beta \in (b_m - b_k)\beta + \langle \alpha, \beta \rangle.\end{aligned}$$

Però també podem reescriure γ com

$$\begin{aligned}\gamma_m &= \lambda\alpha + \mu\beta - \alpha\beta + (\beta - a_k + a_m)\alpha + (b_m - b_k)\beta \\ &= (\beta - a_k + a_m)\alpha + \lambda\alpha + (\mu + b_m - b_k - \alpha)\beta \in (\beta - a_k + a_m)\alpha + \langle \alpha, \beta \rangle.\end{aligned}$$

Per tant, hem demostrat que

$$\{\gamma \in \langle \alpha, \beta \rangle \mid \gamma + i_m - i_k \in \langle \alpha, \beta \rangle\} \subseteq \big((\beta - a_k + a_m)\alpha + \langle \alpha, \beta \rangle\big) \cup \big((b_m - b_k)\beta + \langle \alpha, \beta \rangle\big).$$

Per veure la inclusió contrària, només cal adonar-se de

$$\begin{aligned}(\lambda - a_k + a_m)\alpha + \mu\beta &= \langle \alpha, \beta \rangle, \\ \lambda\alpha + (\mu + b_m - b_k - \alpha)\beta &= \langle \alpha, \beta \rangle.\end{aligned}$$

Aleshores, podem dir que

$$\big(i_k + \langle \alpha, \beta \rangle\big) \cap \big(i_m + \langle \alpha, \beta \rangle\big) = \big(i_m + (\beta - a_k + a_m)\alpha + \langle \alpha, \beta \rangle\big) \cup \big(i_m + (b_m - b_k)\beta + \langle \alpha, \beta \rangle\big).$$

Si ara considerem l'element $i_m + (\beta + a_m - a_k)\alpha$, veiem que pel Lema 6.2.5 que

$$\begin{aligned}
i_m + (\beta + a_m - a_k)\alpha &= \alpha\beta - a_m\alpha - b_m\beta + \alpha\beta + a_m\alpha - a_k\alpha = \alpha\beta - b_m\beta - a_k\alpha + \alpha\beta \\
&= \alpha\beta - b_m\beta - a_k\alpha + \alpha\beta + a_1\alpha - a_1\alpha \\
&= (\beta - a_1)\alpha + (a_1 - a_k)\alpha + (\alpha - b_m)\beta \\
&= j_0 + (a_1 - a_k)\alpha + (\alpha - b_m)\beta \in j_0 + \langle \alpha, \beta \rangle.
\end{aligned}$$

D'altra banda, si considerem l'element $i_m + (b_m - b_k)\beta$, veiem que

$$\begin{aligned}
i_m + (b_m - b_k)\beta &= \alpha\beta - a_m\alpha - b_m\beta + b_m\beta - b_k\beta = \alpha\beta - a_m\alpha - b_k\beta \\
&= \alpha\beta - a_m\alpha - b_k\beta + a_{k+1}\alpha - a_{k+1}\alpha \\
&= \alpha\beta - a_{k+1}\alpha - b_k\beta + (a_{k+1} - a_m)\alpha \\
&= j_k + (a_{k+1} - a_m)\alpha \in j_k + \langle \alpha, \beta \rangle.
\end{aligned}$$

Per tant, $\big(i_k + \langle \alpha, \beta \rangle\big) \cap \big(i_m + \langle \alpha, \beta \rangle\big) \subseteq \big(j_0 + \langle \alpha, \beta \rangle\big) \cup \big(j_k + \langle \alpha, \beta \rangle\big)$. $\square$

Aquest resultat és summament important perquè ens ofereix una forma alternativa d'obtindre la sizígia associada a un semimódul més enllà de la mateixa definició. Més concretament, donat el semimódul Δ generat per I, només cal trobar la seqüència J tal que $[I, J]$ siga un parell fonamental i d'aquesta manera haurem obtingut el sistema minimal de generadors de $\mathrm{Siz}(\Delta)$, que serà J.

Exemple 6.2.7. Siga $S = \langle 5, 7 \rangle$ amb $\mathrm{C}(S) = \{1, 2, 3, 4, 6, 8, 9, 11, 13, 16, 18, 23\}$ i Δ_I el semimódul de S generat per $I = \{0, 8, 6, 9\}$. Aleshores, com

$$\begin{aligned}
8 &= 5 \cdot 7 - 4 \cdot -1 \cdot 7, \\
6 &= 5 \cdot 3 - 4 \cdot -2 \cdot 7, \\
9 &= 5 \cdot 1 - 4 \cdot -3 \cdot 7,
\end{aligned}$$

el semimódul $\mathrm{Siz}(\Delta_I)$ estarà generat minimalment per $J = \{j_0, j_1, j_2, j_3\}$ amb

$$\begin{aligned}
j_0 &= 5 \cdot 7 - 4 \cdot 5 = 35 - 20 = 15, \\
j_1 &= 5 \cdot 7 - 3 \cdot 5 - 1 \cdot 7 = 35 - 15 - 7 = 13, \\
j_2 &= 5 \cdot 7 - 1 \cdot 5 - 2 \cdot 7 = 35 - 5 = 14 = 16, \\
j_3 &= 5 \cdot 7 - 3 \cdot 7 = 35 - 21 = 14.
\end{aligned}$$

Capítol 7

Aportacions a la conjectura de Wilf mitjançant semimóduls

Un dels problemes oberts en la teoria de semigrups numèrics és la conjectura de Wilf. Des de la seua formulació per H. Wilf en [11], només s'han donat respostes parcials i per aquest motiu, és interessant quan es dona la igualtat en la conjectura. En aquest capítol, estudiarem els semimóduls i sizígies d'una família de semigrups numèrics molt concreta amb dimensió d'escabussament màxima, ja que la igualtat de la conjectura ha sigut relacionada amb els semimóduls i sizígies en [12]. Introduirem els conceptes bàsics dels semigrups amb dimensió d'escabussament màxima basant-nos en la secció 1 de [6, Capítol 2]. També farem ús d'algoritmes de càlcul de semimóduls i sizígies implementats en [13] i utilitzant [14] per realitzar experiments numèrics. D'aquesta manera, esperem poder extraure conseqüències sobre aquesta família de semigrups numèrics que ajuden en el procés de demostració de la conjectura de Wilf.

7.1 Càlcul computacional de semimóduls i sizígies

Gràcies a la Proposició 6.1.2, el càlcul de generadors de semimóduls en un semigrup numèric S es redueix a trobar un subconjunt de $\mathrm{C}(S)$ tal que complisca la condició de la mencionada proposició. Una possible aproximació és la de l'Algoritme 2 que permet obtindre semimóduls normalitzats amb el nombre desitjat de generadors minimals.

L'algoritme no és especialment eficient perquè calcula tots els subconjunts possibles de $\mathrm{C}(S)$ i després comprova quins són de la cardinalitat desitjada d'un en un a més de si compleixen la condició de la Proposició 6.1.2 però és adequat per a realitzar els experiments numèrics que mostrarem més endavant. Mitjançant una repetició de l'algoritme, es poden obtindre tots els semimòduls normalitzats de S i una possible implementació de l'algoritme en forma de funció de GAP utilitzant el paquet *numericalsgps* és pot trobar en l'Apèndix B.1.

Algoritme 2: Obtenció del sistema minimal de generadors per a tot semimòdul normalitzat amb n generadors minimals

---

Entrada: El conjunt de clots $\mathrm{C}(S) = \{x_1, \dots, x_r\}$ i $n \in \{1, \dots, r\}$
Eixida : El conjunt $\Omega = \{\mathcal{E}_1^\circ, \dots, \mathcal{E}_l^\circ\}$ amb $\mathcal{E}_i^\circ := \{x_0 = 0, x_{i_1}, \dots, x_{i_{n-1}}\}$ sistema de generadors minimal del semimòdul Δ_i°

```
m := n − 1;
Ω := ∅;
Per a tot ℰ ∈ 𝒫(C(S)) fer
    si |ℰ| = m doncs
        condició := cert;
        Per a tot (x_i, x_j) ∈ (ℰ × ℰ) \ {(x_1, x_1), ..., (x_m, x_m)} fer
            si |x_i − x_j| ∉ C(S) doncs
                condició = fals;
        si condició doncs
            ℰ° := {0} ∪ ℰ;
            Ω = Ω ∪ ℰ°;
```

El concepte de sizígia presentat en la Secció 6.2 es pot generalitzar al context dels semimòduls de semigrups numèrics amb dimensió d'escabussament superior a dos. Així doncs, sent S un semigrup numèric i Δ un semimòdul de S, definiríem la *sizígia* del semimòdul Δ de S com

$$\mathrm{Siz}(\Delta) := \bigcup_{x_i, x_j \in I, i \neq j} \Big((x_i + S) \cap (x_j + S) \Big).$$

Malauradament, el Teorema 6.2.6 no és cert en semigrups numèrics de dimensió d'escabussament superior a dos i per tant s'ha d'utilitzar la definició per a poder calcular la sizígia associada a un semimòdul. Així doncs, la implementació de la definició es presenta com a funció en GAP utilitzant el paquet *numericalsgps* en l'Apèndix B.2.

Exemple 7.1.1. Siga $S = \langle 3, 7, 8 \rangle$. En la Figura 7.1 es mostren els generadors minimals de tots els semimòduls normalitzats de S representats pels seus generadors minimals junt amb els generadors minimals de les seues sizígies associades. També es mostren els generadors minimals de les sizígies normalitzades.

En aquest exemple es veu clarament com no es compleix el Teorema 6.2.6 en semigrups numèrics de dimensió d'escabussament superior a dos. Seguint la notació d'aquest teorema, veiem per exemple que $I = \{0, 1\}$ genera un semimòdul Δ i que $J = \{7, 8, 9\}$ genera $\mathrm{Siz}(\Delta)$. No és possible que $[I, J]$ siguen un parell fonamental perquè, entre altres motius, no tenen ni la mateixa cardinalitat.

```
Semigrup generat per [ 3, 7, 8 ]
[ Semimódul ] --> [ Sizígies ] --> [ Sizígies_normalitzades ]

[ 0 ]
[ 0, 1 ] --> [ 7, 8, 9 ] --> [ 0, 1, 2 ]
[ 0, 2 ] --> [ 8, 9, 10 ] --> [ 0, 1, 2 ]
[ 0, 4 ] --> [ 7, 11, 12 ] --> [ 0, 4, 5 ]
[ 0, 5 ] --> [ 8, 12, 13 ] --> [ 0, 4, 5 ]
[ 0, 1, 2 ] --> [ 7, 8, 9 ] --> [ 0, 1, 2 ]
[ 0, 1, 5 ] --> [ 7, 8, 9 ] --> [ 0, 1, 2 ]
[ 0, 2, 4 ] --> [ 7, 8, 9 ] --> [ 0, 1, 2 ]
[ 0, 4, 5 ] --> [ 7, 8, 12 ] --> [ 0, 1, 5 ]
```

Figura 7.1: Semimóduls i sizígies en $S = \langle 3, 7, 8\rangle$

7.2 Sobre la conjectura de Wilf per a certs semigrups numèrics

Siga S un semigrup numèric, definim el conjunt $\{s \in S \mid s < \mathrm{c}(S)\}$. Anomenem a la cardinalitat d'aquest conjunt el δ-invariant de S, denotada per $\delta(S)$, i està present en la conjectura de Wilf. Així doncs, entenem la conjectura de Wilf com l'afirmació que es compleix

$$\mathrm{c}(S) \geq \mathrm{e}(S) \cdot \delta(S).$$

Aquest és un problema molt ampli i des de la seua formulació, només s'han obtingut resultats per a famílies de semigrups numèrics en particular. Així doncs, és interessant estudiar un problema obert molt relacionat amb aquest que consisteix a afirmar que sols es compleix la igualtat en la conjectura de Wilf per a semigrups numèrics amb $\mathrm{e}(S) = 2$ o de la forma $\langle a, ka+1, Ka+2, \ldots, Ka+(a-1)\rangle$ amb $a, K \in \mathbb{N}$ i $a \geq 2$, veure per exemple [15].

En aquest context resulta útil definir el nombre de Wilf associat a un semigrup numèric S com

$$\mathrm{W}(S) \coloneqq \mathrm{c}(S) - \mathrm{e}(S) \cdot \delta(S).$$

Amb aquesta definició, la conjectura de Wilf es reescriu com $\mathrm{W}(S) \geq 0$ i la igualtat es tradueix en $\mathrm{W}(S) = 0$. Tal com es proposa en [12], també podem definir el número de Wilf associat a un semimódul Δ de S com

$$\mathrm{W}(\Delta) \coloneqq \mathrm{c}(\Delta) - \mathrm{e}(\Delta) \cdot \delta(\Delta),$$

on les diferents invariants es defineixen anàlogament al cas de $\mathrm{W}(S)$. De fet, S és un semimódul de si mateixa. Si Δ està generat minimalment per $[0, x]$ amb $x \in \mathrm{C}(S)$, l'anul·lació de $\mathrm{W}(\Delta)$ està relacionada amb la simetria de Δ, que es defineix d'igual forma que en el cas d'un semigrup numèric, entre altres propietats, veure [12, Theorem 3.4].

Passem a estudiar en més profunditat la família de semigrups numèrics mencionada anteriorment per avançar en l'estudi de la igualtat $\mathrm{W}(S) = 0$. Siguen $a, K \in \mathbb{N}$ amb $a, K \geq 1$, es considera el semigrup

$$S_{a,K} \coloneqq \langle a, Ka + 1, \ldots, Ka + (a-1) \rangle.$$

Aquests semigrups numèrics pertanyen a la família dels semigrups numèrics amb dimensió d'escabussament màxima. Per poder conéixer millor als semigrups numèrics d'aquesta forma, presentem a continuació els coneixements més bàsics dels semigrups numèrics amb dimensió d'escabussament màxima.

Siga S un semigrup numèric. Diguem que S és un semigrup numèric amb *dimensió d'escabussament màxima* si $\mathrm{e}(S) = \mathrm{m}(S)$. Així doncs, podem donar caracteritzacions per aquest tipus de semigrups numèrics en funció de les invariants vistes a la Secció 5.2.

Proposició 7.2.1 ([6], Proposition 3.1)**.** *Siga S un semigrup numèric generat minimalment per $\{a_1 < \cdots < a_e\}$. Aleshores, S és un semigrup numèric amb dimensió d'escabussament màxima si i sols si $Ap(S, a_1) = \{0, a_2, \ldots, a_e\}$.*

Demostració. Siga a_i amb $i = 1, \ldots, e$ un generador minimal de S. Per tant, donat $s \in S^* \setminus \{a_i\}$, tenim que $a_i - s \notin S$ perquè si $a_i - s \in S$, tindríem que $a_i - s + s = a_i$ però açò entra en contradicció amb el fet que $a_i \in S^* \setminus (S^* + S^*)$ per ser generador minimal. Aquest fet implica que $a_i \in \mathrm{Ap}(S, s)$.

Aleshores, deduïm que $\{a_2, \ldots, a_e\} \subseteq \mathrm{Ap}(S, a_1) \setminus \{0\}$. Com la cardinalitat de $\mathrm{Ap}(S, a_1)$ és a_1, tenim que $e = a_1$ si i sols si $\{0, a_2, \ldots, a_e\} = \mathrm{Ap}(S, a_1)$. $\square$

Gràcies a aquest resultat, tenim una caracterització d'un conjunt d'Apéry en els semigrups numèrics amb dimensió d'escabussament màxima i junt amb la Proposició 5.2.2, obtenim els següents resultats.

Corol·lari 7.2.2 ([6], Corollary 3.2)**.** *Siga S un semigrup numèric generat minimalment per $\{a_1 < \cdots < a_e\}$. Aleshores*

1) Si S té dimensió d'escabussament màxima, aleshores $\mathrm{F}(S) = a_e - a_1$,

2) S té dimensió d'escabussament màxima si i sols si $\mathrm{g}(S) = \frac{1}{a_1}(a_2 + \cdots + a_e) - \frac{a_1 - 1}{2}$.

Demostració. Si S té màxima dimensió d'escabussament, aleshores $\mathrm{Ap}(S, a_1) = \{0, a_2, \ldots, a_e\}$.

Per tant, per la Proposició 5.2.2, tenim que

$$\begin{aligned}
\mathrm{F}(S) &= \big(\text{màx Ap}(S,a_1)\big) - a_1 = a_e - a_1,\\
\mathrm{g}(S) &= \frac{1}{a_1}\Big(\sum_{w\in \mathrm{Ap}(S,a_1)} w\Big) - \frac{a_1-1}{2} = \frac{1}{a_1}(a_2+\cdots+a_e) - \frac{a_1-1}{2}.
\end{aligned}$$

□

Note's que el resultat *1)* del corol·lari anterior no és una doble implicació tal com mostra el següent exemple.

Exemple 7.2.3. Siga $S = \langle 4,5,11\rangle$. Tenim que $\mathrm{F}(S) = 11-4 = a_e - a_1$ però S no té dimensió d'escabussament màxima perquè $\mathrm{e}(S) = 3 \neq 4 = \mathrm{m}(S)$.

És fàcil veure que $S_{a,K}$ és un semigrup numèric amb dimensió d'escabussament màxima. Per la seua pròpia definició, tenim que $a, Ka+1, \dots, Ka+(a-1)$ són els seus generadors minimals. Per tant, sabem que $\mathrm{e}(S_{a,K}) = a$ però també $\mathrm{m}(S_{a,K}) = a$ demostrant que $S_{a,K}$ és un semigrup numèric amb dimensió d'escabussament màxima.

A més, pel Corol·lari 7.2.2, sabem que $\mathrm{c}(S_{a,K}) = Ka+(a-1)-a+1 = Ka$. També tenim que

$$\begin{aligned}
\mathrm{g}(S_{a,K}) &= \frac{1}{a}(Ka+1+\cdots+Ka+(a-1)) - \frac{a-1}{2}\\
&= \frac{1}{a}(Ka(a-1)+1+\cdots+(a-1)) - \frac{a-1}{2}\\
&= \frac{1}{a}\left(Ka(a-1)+\frac{a(a-1)}{2}\right) - \frac{a-1}{2} = K(a-1)+\frac{a-1}{2}-\frac{a-1}{2} = K(a-1)
\end{aligned}$$

També és fàcil comprovar que $\delta(S) = \mathrm{c}(S) - \mathrm{g}(S)$ per a qualsevol semigrup numèric S i per tant $\delta(S_{a,K}) = Ka - (K(a-1)) = Ka - Ka + K = K$.

Exemple 7.2.4. En la Figura 7.2 es mostren tots els semimóduls del semigrup numèric $S_{4,2}$ representats pels seus generadors minimals. A més, apareixen les sizígies associades i normalitzades en funció dels seus generadors minimals també. Tanmateix, es representa per a cada semimódul el seu nombre de Wilf i la cardinalitat dels conjunts de semimóduls $\mathrm{G}x$, on $\mathrm{G}x$ representa el conjunt de semimóduls amb dimensió d'escabussament x.

A la vista de l'Exemple 7.2.4 i l'Exemple 7.1.1, que descriuen els semigrups numèrics $S_{4,2}$ i $S_{3,2}$ respectivament, així com de més proves numèriques realitzades utilitzant l'Apèndix B.4, es veu clarament que els clots de $S_{a,K}$ es distribueixen amb una periodicitat concreta. Tot seguit els caracteritzarem i descriurem en una nova estructura.

```
Semigrup generat per [ 4, 9, 10, 11 ]
( Nombre_Wilf ) [ Semimóduls ] --> [ Sizígies ] --> [ Sizígies_normalitzades ]
Semimòduls G0
[ 0 ]
Cardinal G0 = 1

Semimòduls G1
( 0 ) [ 0, 1 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( 0 ) [ 0, 2 ] --> [ 10, 11, 12, 13 ] --> [ 0, 1, 2, 3 ]
( 1 ) [ 0, 3 ] --> [ 11, 12, 13, 14 ] --> [ 0, 1, 2, 3 ]
( 2 ) [ 0, 5 ] --> [ 9, 14, 15, 16 ] --> [ 0, 5, 6, 7 ]
( 2 ) [ 0, 6 ] --> [ 10, 15, 16, 17 ] --> [ 0, 5, 6, 7 ]
( 3 ) [ 0, 7 ] --> [ 11, 16, 17, 18 ] --> [ 0, 5, 6, 7 ]
Cardinal G1 = 6

Semimòduls G2
( -10 ) [ 0, 1, 2 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -8 ) [ 0, 1, 3 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -7 ) [ 0, 1, 6 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -5 ) [ 0, 1, 7 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -6 ) [ 0, 2, 3 ] --> [ 10, 11, 12, 13 ] --> [ 0, 1, 2, 3 ]
( -7 ) [ 0, 2, 5 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -3 ) [ 0, 2, 7 ] --> [ 10, 11, 12, 13 ] --> [ 0, 1, 2, 3 ]
( -5 ) [ 0, 3, 5 ] --> [ 9, 11, 12, 14 ] --> [ 0, 2, 3, 5 ]
( -3 ) [ 0, 3, 6 ] --> [ 10, 11, 12, 13 ] --> [ 0, 1, 2, 3 ]
( -4 ) [ 0, 5, 6 ] --> [ 9, 10, 15, 16 ] --> [ 0, 1, 6, 7 ]
( -2 ) [ 0, 5, 7 ] --> [ 9, 11, 14, 16 ] --> [ 0, 2, 5, 7 ]
( 0 ) [ 0, 6, 7 ] --> [ 10, 11, 16, 17 ] --> [ 0, 1, 6, 7 ]
Cardinal G2 = 12

Semimòduls G3
( 0 ) [ 0, 1, 2, 3 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -8 ) [ 0, 1, 2, 7 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -5 ) [ 0, 1, 3, 6 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -4 ) [ 0, 1, 6, 7 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -2 ) [ 0, 2, 3, 5 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -4 ) [ 0, 2, 5, 7 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( -1 ) [ 0, 3, 5, 6 ] --> [ 9, 10, 11, 12 ] --> [ 0, 1, 2, 3 ]
( 0 ) [ 0, 5, 6, 7 ] --> [ 9, 10, 11, 16 ] --> [ 0, 1, 2, 7 ]
Cardinal G3 = 8

Suma cardinals = 27
Enter x = 3
```

Figura 7.2: Semimóduls, sizígies i nombres de Wilf en $S_{4,2} = \langle 4, 9, 10, 11 \rangle$

Sabem que $S_{a,K}$ és un semigrup numèric amb dimensió d'escabussament màxima i per tant $\mathrm{Ap}(S_{a,K}, a) = \{0, Ka+1, \ldots, Ka+(a-1)\}$. Estudiem millor el que representa cada element de $\mathrm{Ap}(S_{a,K}, a)$. Tenim que $Ka+i \in \mathrm{Ap}(S_{a,K}, a)$ per a $i = 1, \ldots, a-1$. Pel Lema 5.1.5, resulta que $Ka+i$ és el menor element de $S_{a,K}$ congruent amb i módul a. Per tant, $(K-j)a+i \in \mathrm{C}(S_{a,K})$ per a $j = 1, \ldots, K$. És a dir, que $Ka+i$ és el representant d'una classe modular a que conté K clots. Com $\mathrm{g}(S_{a,k}) = K(a-1)$, deduïm que tots els clots de $S_{a,k}$ estan continguts en aquestes classes. Anomenarem *forat* al conjunt $\mathcal{F}_{K-j+1} = \{(K-j)a+1, (K-j)a+2 \ldots, (K-j)a+(a-1)\}$ amb $j = \{j = 1, \ldots, K\}$. La seua relació amb $\mathrm{Ap}(S_{a,k}, a)$ es veu gràficament en l'Exemple 7.2.6 i tota aquesta anàlisi deriva en el següent resultat.

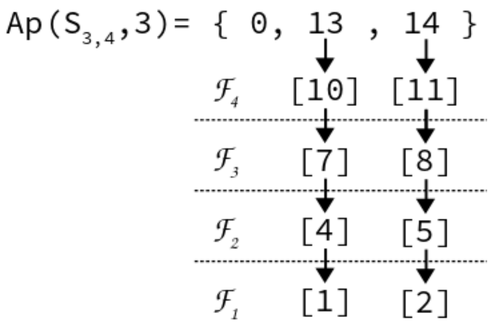
Proposició 7.2.5. *Siga $S_{a,K}$ un semigrup numèric. Tots els clots de $S_{a,K}$ es distribueixen en K forats $\mathcal{F}_i$ per a $i = 1, \ldots, K$ com*

- *forat entre* 0 *i* a, $\mathcal{F}_1 = \{1, 2, \ldots, a-1\}$,
- *forat entre* a *i* $2a$, $\mathcal{F}_2 = \{a+1, a+2, \ldots, 2a-1\}$,
- *forat entre* $2a$ *i* $3a$, $\mathcal{F}_3 = \{2a+1, 2a+2, \ldots, 3a-1\}$,
- $\vdots$
- *forat entre* $(K-1)a$ *i* Ka, $\mathcal{F}_K = \{(K-1)a+1, (K-1)a+2, \ldots, Ka-1\}$.

on el cardinal de cada forat $\mathcal{F}_i$ és exactament $a-1$,

En funció de la pertinença a un forat o un altre, denotem als clots com $f_{i,j}$ per al clot en la posició j del forat $\mathcal{F}_i$, això és, que $f_{i,j} = (i-1)a+j$ per a $i = 1, \ldots, K$ i $j = 1, \ldots, a_1$.

Exemple 7.2.6. Siga $S_{3,4} = \langle 3, 13, 14\rangle$. Aleshores, tenim que $\mathrm{C}(S_{3,4}) = \{1, 2, 4, 5, 7, 8, 10, 11\}$ i $\mathrm{Ap}(S_{3,4}, 3) = \{0, 13, 14\}$.



$$\begin{array}{ll} (4-1)\cdot 3+1=10 & (4-1)\cdot 3+2=11 \\ (4-2)\cdot 3+1=7 & (4-2)\cdot 3+2=8 \\ (4-3)\cdot 3+1=4 & (4-3)\cdot 3+2=5 \\ (4-4)\cdot 3+1=1 & (4-4)\cdot 3+2=2 \end{array}$$

En relació amb la conjectura de Wilf, interessa estudiar el δ-invariant i el conductor dels semimóduls Δ_I de $S_{a,K}$ generats minimalment per $I = [0, x]$ amb $x \in \mathrm{C}(S)$. És a dir, els semimóduls del conjunt de semimóduls G1 definit en l'Exemple 7.2.4 i l'Exemple 7.1.1. Així doncs, donem les expressions del δ-invariant i el conductor en aquests semimóduls.

Proposició 7.2.7. *Siga x un clot de $S_{a,K}$. Si $I = [0, x]$, aleshores el conductor del semimódul Δ_I de $S_{a,K}$ és*

$$\mathrm{c}(\Delta_I) = \begin{cases} Ka = \mathrm{c}(S_{a,K}), & si \;\; x = f_{i,j} \; \forall \; i = 1, \ldots, K \;\, i \, \forall \; j \in \{1, \ldots, a-2\}, \\ Ka - 1, = \mathrm{c}(S_{a,K}) - 1 & si \;\; x = f_{i,a-1} \; per \; a \; qualsevol \; i = 1, \ldots, K. \end{cases}$$

Demostració. Siga $f_{i,j} \in \mathrm{C}(S_{a,K})$ per a $i = 1, \ldots, K$ i $j = 1, \ldots, a-1$. Si $I = [0, f_{i,j}]$ és el sistema minimal de generadors del semimódul Δ_I, tenim que

$$\Delta_I = S_{a,K} \cup \{(i-1)a + j, (i-2)a + j, \ldots, (i-K)a + j\} = S_{a,K} \cup \{(n-1)a + j\}_{n=i}^{K}.$$

És a dir, que Δ_I està format per tots els elements de $S_{a,K}$ i dels clots de $S_{a,K}$ majors que $f_{i,j}$ que siguen congruents amb j módul a o dit d'altra manera, tots els elements en la posició j dels forats $\mathcal{F}_m$ amb $m \in \{i, i+1, \ldots, K\}$.

Per definició de semimódul, tenim que $\Delta_I + S_{a_K} \subseteq \Delta_I$ i per tant no és difícil deduïr que $\mathrm{c}(\Delta_I) \leq \mathrm{c}(S_{a,K})$. Com $\mathrm{F}(S_{a,K}) = Ka - 1$, veiem que

$$\mathrm{F}(S_{a,K}) = Ka - 1 = Ka - 1 + a - a = (K-1)a + (a-1) \in \{(n-1)a + (a-1)\}_{n=i}^{K}$$

Per tant, si $j = a - 1$, tenim que $\mathrm{F}(S_{a,K}) \in \Delta_I$ i conseqüentment $\mathrm{c}(\Delta_I) = \mathrm{c}(S_{a,K}) - 1$. D'altra banda, si $j \neq a - 1$, tenim que $\mathrm{F}(S_{a,K}) \notin \Delta_I$ deduïnt que $\mathrm{c}(\Delta_I) = \mathrm{c}(S_{a,K})$. □

Proposició 7.2.8. *Siga $x \in C(S_{a,K})$ i siga Δ_I el semimódul de $S_{a,K}$ corresponent a $I = [0, x]$. Aleshores, el δ-invariant de Δ_I és*

$$\delta(\Delta_I) = \begin{cases} 2K - i + 1, & si \;\; x = f_{i,j} \; \forall \; i = 1, \ldots, K \;\, i \, \forall \; j \in \{1, \ldots, a-2\}, \\ 2k - i, & si \;\; x = f_{i,a-1} \; per \; a \; qualsevol \; i = 1, \ldots, K. \end{cases}$$

Demostració. Siga $f_{i,j} \in \mathrm{C}(S_{a,K})$ per a $i = 1, \ldots, K$ i $j = 1, \ldots, a-1$. Si $I = [0, f_{i,j}]$, tenim $\Delta_I = S_{a,K} \cup \{(n-1)a + j\}_{n=i}^{K}$. Per definició, sabem que $\delta(\Delta_I) = |\{s \in \Delta_I \mid s < \mathrm{c}(\Delta_I)\}|$ i per tant, tenim que

$$\begin{aligned}\delta(\Delta_I) &= |\{s \in \Delta_I \mid s < \mathrm{c}(\Delta_I)\}| \\ &= |\{s \in S_{a,K} \cup \{(n-1)a + j\}_{n=i}^{K} \mid s < \mathrm{c}(\Delta_I)\}| \\ &= |\{s \in S_{a,K} \mid s < \mathrm{c}(\Delta_I)\}| + |\{s \in \{(n-1)a + j\}_{n=i}^{K} \mid s < \mathrm{c}(\Delta_I)\}|.\end{aligned}$$

Per un costat, si $j \neq a-1$, aleshores $\mathrm{c}(\Delta_I) = \mathrm{c}(S_{a,k}) = Ka$ per la Proposició 7.2.7 i $|\{s \in \{(n-1)a+j\}_{n=i}^{K} \mid s < Ka\}| = K+i-1$ perquè $(K-1)a+j < Ka$ per a $j = 1, \ldots, a-1$ així que

$$\begin{aligned}\delta(\Delta_I) &= |\{s \in S_{a,K} \mid s < Ka\}| + |\{s \in \{(n-1)a + j\}_{n=i}^{K} \mid s < Ka\}| \\ &= \delta(S_{a,K}) + K - i + 1 = 2K - i + 1.\end{aligned}$$

Per l'altre costat, si $j = a-1$, aleshores $\mathrm{c}(\Delta_I) = \mathrm{c}(S_{a,k}) - 1 = Ka - 1$ però

$$|\{s \in S_{a,K} \mid s < Ka - 1\}| = |\{s \in S_{a,K} \mid s < Ka\}|$$

perquè $\mathrm{F}(S_{a,K}) = Ka-1$. També tenim que $|\{s \in \{(n-1)a+(a-1)\}_{n=i}^{K} \mid s < aK-1\}| = K-i$ perquè $(K-1)a+j < Ka-1$ per a $j = 1, \ldots, a-2$, ja que $(K-1)a+(a-1) = Ka-a+a-1 = Ka-1$. Així doncs, tenim que

$$\begin{aligned}\delta(\Delta_I) &= |\{s \in S_{a,K} \mid s < Ka - 1\}| + |\{s \in \{(n-1)a + j\}_{n=i}^{K} \mid s < Ka - 1\}| \\ &= |\{s \in S_{a,K} \mid s < Ka\}| + K - i = \delta(S_{a,k}) + K - i = 2K - i.\end{aligned}$$

□

Exemple 7.2.9. Prenguem el semigrup numèric $S_{4,3} = \langle 4, 13, 14, 15\rangle$. Aleshores, tenim que $\mathrm{C}(S) = \{1,2,3,5,6,7,9,10,11\}$. En aquest semigrup es distingueixen $K = 3$ forats, que són $\mathcal{F}_1 = \{1,2,3\}$, $\mathcal{F}_2 = \{5,6,7\}$ i $\mathcal{F}_3 = \{9,10,11\}$. La següent taula presenta els semimóduls generats per $[0,x]$ amb $x \in \mathrm{C}(S_{4,3})$ junt amb els seus corresponents conductor i δ-invariant.

$I = [0,x]$	$\mathrm{c}(\Delta_I)$	$\delta(\Delta_I)$
$[0,1]$	$12 = 3 \cdot 4$	$6 = 2 \cdot 3$
$[0,2]$	$12 = 3 \cdot 4$	$6 = 2 \cdot 3$
$[0,3]$	$11 = 3 \cdot 4 - 1$	$5 = 2 \cdot 3 - 1$
$[0,5]$	$12 = 3 \cdot 4$	$5 = 2 \cdot 3 - 1$
$[0,6]$	$12 = 3 \cdot 4$	$5 = 2 \cdot 3 - 1$
$[0,7]$	$11 = 3 \cdot 4 - 1$	$4 = 2 \cdot 3 - 2$
$[0,9]$	$12 = 3 \cdot 4$	$4 = 2 \cdot 3 - 2$
$[0,10]$	$12 = 3 \cdot 4$	$4 = 2 \cdot 3 - 2$
$[0,11]$	$11 = 3 \cdot 4 - 1$	$3 = 2 \cdot 3 - 3$

A partir dels resultats anteriors, som capaços de presentar un nou resultat relacionat amb el nombre de Wilf en aquest context.

Corol·lari 7.2.10. *Si* x_1, x_2 *clots de* $S_{a,K}$ *amb* $x_1 > x_2$ *i* $I_1 = [0, x_1]$*,* $I_2 = [0, x_2]$ *són els sistemes de generadors minimals dels semimóduls* Δ_{I_1} *i* Δ_{I_2} *respectivament, aleshores,* $\mathrm{W}(\Delta_{I_1}) \geq \mathrm{W}(\Delta_{I_2})$.

Demostració. Siga $x \in \mathrm{C}(S_{a,K})$ i siga Δ_I el semimódul de $S_{a,K}$ corresponent a $I = [0, x]$. Per definició, tenim que

$$\mathrm{W}(\Delta_I) = \mathrm{c}(\Delta_I) - \mathrm{e}(\Delta_I).\cdot \delta(\Delta_I)$$

Per la Proposició 7.2.7 i la Proposició 7.2.8, sent $d_1, d_2 \in \{0, 1\}$, podem escriure

$$\begin{aligned} \mathrm{W}(\Delta_I) &= 2K - d_1 - 2(2K - i + d_2) \\ &= 2Kd_1 - 4K + 2i - 2d_2 \\ &= -2K - d_1 + 2i - 2d_2. \end{aligned}$$

Però, tenim que $d_1 = 0$ si i sols si $d_2 = 1$ i $d_1 = 1$ si i sols si $d_2 = 0$. Per tant, $\mathrm{W}(\Delta_I)$ només admet dos posibilitats:

$$\mathrm{W}(\Delta_I) = -2K + 2i - 2 \quad \text{ó} \quad \mathrm{W}(\Delta_I) = -2K + 2i - 1.$$

Siguen $I_1 = [0, x_1]$, $I_2 = [0, x_2]$ els sistemes de generadors minimals dels semimóduls Δ_{I_1} i Δ_{I_2} respectivament on $x_1, x_2 \in \mathrm{C}(S_{a,K})$, si denotem $x_1 = f_{i_1,j_1}$ i $x_2 = f_{i_2,j_2}$, és fàcil veure que si $x_1 > x_2$, aleshores $i_1 \geq i_2$ i per tant $\mathrm{W}(\Delta_{I_1}) > \mathrm{W}(\Delta_{I_2})$. □

Així doncs, podem veure un exemple d'aquest resultat en la Figura 7.2. Encara que siga un resultant interessant, ho és més la demostració del recíproc del Corol·lari 7.2.10. Aquest resultat implicaria que les úniques successions de nombres de Wilf de semimóduls creixents que podem trobar són les que tenim al treballar amb semimóduls de dimensió d'escabussament dos dels semigrups $S_{a,K}$. Presentem aquest fet com una conjectura, ja que implica un estudi més enllà de la intenció d'aquest treball.

Capítol 8

Conclusions

Al llarg de tot el treball hem partit de la noció de semigrup numèric i hem anat presentant resultats cada vegada més específics. En aquest sentit, hem presentat primer que tot semigrup numèric està únicament i finitament generat i hem vist que dues de les seues invariants més importants, com són el nombre de Frobenius i el gènere, venen determinades en funció del conjunt d'Apéry, demostrant que és una ferramenta essencial quan es treballa amb semigrups numèrics.

A continuació, hem presentat les nocions de semimóduls i sizígies i hem descrit exactament la seua relació, fent ús dels parells fonamentals, per a semigrups numèrics amb dimensió d'escabussament dos passant per demostrar quina és l'estructura dels clots d'aquests semigrups numèrics.

Finalment, hem desenvolupat ferramentes computacionals per fer proves numèriques i trobar comportaments que puguem descriure formalment esperant que aquestes descripcions formals ajuden en la investigació de la conjectura de Wilf.

Així doncs, el treball presenta una estructura que representa la investigació matemàtica com a concepte, perquè s'ha començat exposant resumidament els resultats d'una font molt contrastada i assentada en la comunitat científica per a continuar fent el mateix però amb una investigació avantguardista i finalment experimentant d'acord amb aquestes investigacions generant nous resultats que aportar a la comunitat científica.

En l'àmbit personal, aquesta investigació que ha nascut del coneixement més assentat i ha acabat en la generació d'experiments i nous resultats, ha sigut una de les coses que valore més positivament del treball i pense que ha afavorit el meu desenvolupament com a investigador matemàtic.

Apèndix A

Informe del algoritmo de estimación del tiempo de espera en colas

El algoritmo de estimación está pensado para que se ejecute para cada usuario en una cola cuando entra al sistema (se apunta a la cola) o cuando su posición en la cola cambia (se le completa el servicio a un usuario o alguien por delante abandona la cola). Por tanto, la entrada del algoritmo serán tres: el identificador del usuario al que se le va a realizar la estimación, el identificador de la cola en la que se encuentra y el *timestamp* del momento en que se inicia el algoritmo.

A partir de estos tres datos, se calcularán otros tres datos (personas en el sistema, número de servidores y tasa de servicio) que serán los parámetros para la función que efectivamente calculará la estimación del tiempo de espera. La eficacia de la estimación radicará en obtener rápidamente estos valores y especialmente en calcular una tasa de servicio media lo más ajustada a la situación real en la que se va a realizar la estimación.

Personas en el sistema

Obtener el número de personas por delante en la cola del usuario al que se la realiza la estimación debería ser una sencilla consulta a la base de datos ya que se tiene el identificador de la cola en la que el usuario va a esperar. En la base de datos se deberá mantener actualizada una lista de usuarios esperando en cada cola (los que están esperando y a los que se está sirviendo) y bastará con restarle 1 al tamaño de esa lista para obtener el número de personas total en el sistema (el usuario al que se realiza la estimación también contabiliza como dentro del sistema y de ahí la resta de 1).

Número de servidores

Para obtener el número de servidores que están atendiendo en el momento de realizar la estimación será suficiente con una consulta sencilla en la línea del anterior párrafo. Con el identificador de cola en la que el usuario va a esperar, se debería poder obtener el número de servidores activos atendiendo en esa cola en el momento en el que se calcula la estimación. Como en el caso anterior, será necesario mantener una lista actualizada de los servidores atendiendo en cada cola.

Tasa de servicio

Este parámetro es el más complejo de calcular ya que hay que calcular la media del tiempo que pasa cada servidor atendiendo a cada usuario en la cola. Después bastará con calcular la media de las medias de cada servidor para obtener un tasa de servicio única por cada cola aunque haya más de un servidor atendiendo en ella.

Primero, al disponer del identificador de la cola en la que el usuario esperará, se puede obtener qué servidores están atendiendo en ese momento en la cola. Una vez sabiendo los identificadores de los servidores que están atendiendo en la cola, se accederá a su registro histórico de servicios seleccionando los servicios que haya realizado previamente en la cola en la que está atendiendo en el momento de la estimación y además que cumpla con los parámetros que describen la situación de la cola en el momento de calcular la estimación (intervalo horario del día, día de la semana, en temporada de rebajas o no, con fiestas en la localidad del comercio o no, etc).

Con los subconjuntos seleccionados (según las condiciones anteriores) de cada servidor activo en la cola, se calculará una media para cada servidor del tiempo que tardó en atender a los usuarios quitando previamente el 30% de los datos más altos (dejar todos los servicios realizados por debajo del percentil/quantil 70). Por último, se realizará una media de todas esas medias para reducir a un único valor la tasa de servicio en la cola en la que se realizará la estimación.

Cálculo de la estimación

Una vez obtenidos los tres parámetros mencionados anteriormente, el cálculo de la estimación es muy sencillo. Primero, se resta al número de personas en el sistema ($numUsersSystem$) el número de servidores ($numServers$) para obtener el número de personas esperando para ser atendidas por delante del usuario al que se le calcula la estimación ($numQueuersAhead$).

A continuación, se evalúa si el número máximo de personas a las que no se les va a realizar la estimación y deberán dirigirse al comercio porque serán atendidas en breve (*numNextServedQueuers*) es menor o igual que el número de personas por delante del usuario al que se le calcula la estimación. Es decir, cada comercio podrá fijar un número que representará el número máximo de personas que podrán estar esperando físicamente en el punto de servicio.

En caso afirmativo de la comparación, se interpreta que el punto de servicio ya tiene el máximo número de personas esperando allí físicamente y se devolverá al usuario el producto entre la tasa de servicio previamente calculada (*serviceTime*) y el número de personas esperando a ser atendidas por delante del usuario al que se le realiza la estimación.

En caso contrario, se interpreta que todavía no se ha llegado a ese de personas esperando físicamente en el punto de servicio y por tanto se devuelve un valor nulo que se traducirá en devolverle al usuario un mensaje indicando que se dirija hacia el comercio para ser atendido en breve.

```
class WaitingTimeEstimation {
    execute(numUsersSystem: number, numServers: number, numNextServed Queuers:
    number, serviceTime: number) : number {
        let numQueuersAhead: number = numUsersSystem - numServers -1;
        if (numNextServedQueuers <= numQueuersAhead) {
            return serviceTime * numQueuersAhead;
        }
        return null;
    }
}
```

Figura A.1: Posible implementación en TypeScript de la función que calcula el tiempo estimado de espera en la cola.

Consideraciones finales

En el algoritmo hay varios parámetros que pueden (y deben) ser ajustados en base a los datos reales. Uno de ellos es eliminar el 30% de los datos más altos al calcular la tasa de servicio. Ese 30% es un valor fijado a priori pero dependiendo del comercio puede funcionar mejor cambiarlo por un 20%, 10% o incluso 40% hasta encontrar el valor que ofrezca una estimación satisfactoria.

Otro valor a ajustar es el conjunto de parámetros que describen el entorno de una cola. En el algoritmo se proponen algunos parámetros lógicos (hora o día de la semana) pero pueden diferir cuáles de ellos aportan una mejora a la estimación en función del comercio y también deberán ser ajustados respecto a la cantidad de datos (por ejemplo, cuantos más hayan más pequeños se podrán hacer los intervalos horarios).

Una posible forma de optimizar la estimación sería crear conjuntos de estos parámetros y periódicamente utilizar el histórico de servicios para generar una estimación de tiempo en la llegada de los usuarios y compararla con el tiempo real que tuvieron que esperar. El conjunto que ofrezca una diferencia más pequeña será el que mejor estimaciones produzca.

Apèndix B

Codi dels programes utilitzats

En aquest apèndix presentarem el codi dels programes que ens han permés realitzar totes les proves numèriques tanmateix com les implementacions dels diferents algoritmes presentats al llarg del treball.

B.1 SemimodulesGeneratorsOfNumericalSemigroup

```
SemimodulesGeneratorsOfNumericalSemigroup := function(numerical_semigroup, semimodules_generators_nozero_cardinal)
  local gaps, subsets_gaps, list;
  gaps := Gaps(numerical_semigroup);
  if semimodules_generators_nozero_cardinal = 0 then
    return [[0]];
  elif semimodules_generators_nozero_cardinal = 1 then
    return List(gaps, x -> Concatenation([0], [x]));
  else
    subsets_gaps := Combinations(gaps, semimodules_generators_nozero_cardinal);
    list := Filtered(subsets_gaps,x->ForAll(Combinations(x,2),y->AbsInt(y[1]-y[2]) in gaps));
    return List(list, x -> Concatenation([0], x));
  fi;
end;
```

B.2 SyzygyByGeneratorsSemimoduleOfNumericalSemigroup

```
SyzygyByGeneratorsSemimoduleOfNumericalSemigroup := function(numerical_semigroup, generators_semimodule)
  local generators_syzygy, couple_generators;
  generators_syzygy := [];
  for couple_generators in Combinations(generators_semimodule, 2) do
    Append(generators_syzygy, MinimalGenerators(Intersection(couple_generators[1]+numerical_semigroup, couple_generators[2] + numerical_semigroup)));
  od;
  return generators_syzygy + numerical_semigroup;
end;
```

B.3 NumberWilf

```
NumberWilf := function(semigroup_generators, semimodule_generators)
  local s, i;
  s := NumericalSemigroup(semigroup_generators);
  i := semimodule_generators + s;
  return Conductor(i) - (Length(MinimalGenerators(i)) * (Length(SmallElements(i))-1));
end;
```

B.4 Programa principal de generació de proves numèriques

```
numerical_semigroups:= []; k := 2;
for a in [2 .. 7] do
  gents := [a];
  for i in [1 .. a-1] do
    Add(gents, k*a+i);
  od;
  Add(numerical_semigroups, NumericalSemigroup(gents));
od;
name := Filename( Directory("/home/alfredo/TFG/Output/"), "cuentas_k-2.txt");
output := OutputTextFile(name, false);
AppendTo(output, "----------------\n");
for numerical_semigroup in numerical_semigroups do
  AppendTo(output, "\n");
  gen := MinimalGenerators(numerical_semigroup);
  AppendTo(output, "\n", "Semigrup generat per ", gen, "\n");
  sum_cardinal := 0; x := 0;
  g := SemimodulesGeneratorsOfNumericalSemigroup(numerical_semigroup, x);
  while g <> [] do
    AppendTo(output, "Semimòduls G", x, "\n");
    if g <> [[0]] then
      list_wilf_numbers := []; list_syzygies_generators := [];
      for semimodule_generators in g do
        wilf_number := NumberWilf(gen, semimodule_generators);
        AppendTo(output, "( ", wilf_number, " ) ");
        AppendTo(output, semimodule_generators, " --> ");
        syzygy := SyzygyByGeneratorsSemimoduleOfNumericalSemigroup(
    numerical_semigroup, semimodule_generators);
        AppendTo(output, MinimalGenerators(syzygy), " --> ");
        syzygy_norm_generators := List(MinimalGenerators(syzygy), x -> x -
    Minimum(MinimalGenerators(syzygy)));
        AppendTo(output, syzygy_norm_generators, "\n");
      od;
    else
      AppendTo(output, g[1], "\n");
    fi;
    cardinal := Length(g);
    AppendTo(output, "Cardinal G", x, " = ", cardinal, "\n\n");
    sum_cardinal := sum_cardinal + cardinal;
    x := x + 1;
    g := SemimodulesGeneratorsOfNumericalSemigroup(numerical_semigroup, x);
  od;
  AppendTo(output, "Suma cardinals = ", sum_cardinal, "\n");
  AppendTo(output, "Enter x = ", x-1, "\n\n");
  AppendTo(output, "----------------\n");
od;
CloseStream(output);
```

Bibliografia

[1] Abad, Ricardo Cao: *Introducción a la simulación y a la teoría de Colas*. Netbiblo, 2002.

[2] *Intro to data structures*. https://pandas.pydata.org/pandas-docs/stable/user_guide/dsintro.html. Accedit: 14-09-2020.

[3] *The MongoDB 4.4 Manual*. https://docs.mongodb.com/manual. Accedit: 14-09-2020.

[4] *TypeScript Handbook*. https://www.typescriptlang.org/docs/handbook/intro.html. Accedit: 14-09-2020.

[5] *TypeScript Deep Dive*. https://basarat.gitbook.io/typescript. Accedit: 14-09-2020.

[6] Rosales, J. C. i P. A. García-Sánchez: *Numerical Semigroups*, volum 20 de *Developments in Mathematics*. Springer, 2009, ISBN 978-1-4419-0159-0.

[7] Ilhan, Sedat: *On Apéry Sets of Symmetric Numerical Semigroups*. International Mathematical Forum, 1(10):481–484, 2006.

[8] Moyano-Fernández, Julio José i Jan Uliezka: *Lattice paths with given number of turns and semimodules over numerical semigroups*. Semigroup Forum, 88(3):632–633, 2014.

[9] Rosales, J. C.: *Fundamental gaps of numerical semigroups generated by two elements*. Linear Algebra and its Applications, 405:200–208, 2005.

[10] Moyano-Fernández, Julio José i Jan Uliezka: *Hilbert depth of graded modules over polynomial rings in two variables*. Journal of Algebra, 373:130–152, 2013.

[11] Wilf, H. S.: *A circle-of-lights algorithm for the money-changing problem*. The American Mathematical Monthly, 85(7):562–565, 1978.

[12] Moyano-Fernández, J. J. i P. Almirón: *Supersymmetric gaps of a numerical semigroup with two generators*. 2020.

[13] The GAP Group: *GAP – Groups, Algorithms, and Programming, Version 4.11.0*, 2020. https://www.gap-system.org.

[14] Delgado, M., P. A. Garcia-Sanchez i J. Morais: *NumericalSgps, A package for numerical semigroups, Version 1.2.2*. https://gap-packages.github.io/numericalsgps, Mar 2020. Refereed GAP package.

[15] Moscarrielloa, A. i A. Sammartano: *On a conjecture of Wilf about the Frobenius number*. Math. Z., 280:47–53, 2015.