Local Symmetry Preserving Operations on Polyhedra

Pieter Goetschalckx

Submitted to the Faculty of Sciences of Ghent University in fulfilment of the requirements for the degree of Doctor of Science: Mathematics.

GHENT
UNIVERSITY

Supervisors
prof. dr. dr. Kris Coolsaet
dr. Nico Van Cleemput

Chair
prof. dr. Marnix Van Daele

Examination Board
prof. dr. Tomaž Pisanski
prof. dr. Jan De Beule
prof. dr. Tom De Medts
dr. Carol T. Zamfirescu
dr. Jan Goedgebeur

Department of Applied Mathematics, Computer Science and Statistics
Faculty of Sciences, Ghent University

In memory of
John Horton Conway
(1937–2020)

Contents

List of tables

Dankwoord

Welkom in mijn doctoraatsthesis, die ondanks mijn voorliefde voor het nipt halen van deadlines dan toch op tijd is afgeraakt. Dat is zeker niet enkel mijn verdienste, dus eerst en vooral moeten er enkele mensen bedankt worden.

Mama en papa, zonder jullie zou ik — en bijgevolg dit doctoraat — er niet geweest zijn. Ook de kans om in Antwerpen wiskunde te gaan studeren, en er daarna in Gent nog een master wiskundige informatica bij te doen, heb ik aan jullie te danken. Jullie hadden toen waarschijnlijk geen vermoeden dat die verhuis naar Gent achteraf redelijk definitief zou blijken. Gelukkig verkleinen jullie die afstand met de vele familieweekends, uitstappen, feesten en bezoeken aan Gent. Nu we over familie bezig zijn, verdienen ook mijn broers (en hun partners) een eervolle vermelding. Bedankt Andries om er op het nippertje voor te zorgen dat ik de laatste zal zijn om "echt" werk te vinden, met een contract in plaats van een statuut enzo. Als je nu nog trouwt met Louise, een huis koopt en een rijbewijs haalt, ben ik met alles de laatste. Bedankt Siemen en Michiel om voor een jaar onze overburen te zijn geweest, waardoor we van vele treinreizen naar Antwerpen gespaard zijn gebleven. Rijden jullie vanaf nu even om langs Gent? Bedankt Michiel en Eline voor Sep en Raf, die de verplaatsing naar Antwerpen altijd de moeite waard maken en wanneer dat niet kan door coronamaatregelen voor entertainment zorgen tijdens het "met alle mensen bellen". Natuurlijk kan ook de bomma niet onvermeld blijven. Bedankt voor de bezoekjes aan Gent, de onverwachte telefoontjes en het jeugdig enthousiasme.

Mijn medestudenten in de bachelor wiskunde, in het bijzonder Alicia, Dorien, Joachim, Lynn, Pieter, Roy, Kevin en Thomas wil ik vooral bedanken om hun mondelinge examens in de voormiddag te plannen zodat ik kon uitslapen tot de namiddag, maar ook voor de extracurriculaire activiteiten. Bovendien zijn die eerste zes zo moedig geweest om voor de masteropleiding op verkenning naar Gent te gaan, zodat ik twee jaar later gerust kon zijn dat het daar veilig was. Gelukkig was ik niet de enige die in Antwerpen achterbleef. Kevin, bedankt om het nog twee jaar langer met mij uit te houden. Lynn en Roy zijn daarna met een missie begonnen om alle assistenten in gebouw S9 door Antwerpenaren te vervangen, waar ik me later bij heb aangesloten, maar dat is niet helemaal gelukt.

In mijn eerste jaar in Gent leerde ik dat informatici in tegenstelling tot wiskundigen niet naar de les gaan, had ik les van en met enkele toekomstige collega's, en ontdekte ik dat samenvallende lessen een goed excuus zijn om naar geen van beide te gaan. Maar Gunnar zijn lessen *Algoritmische grafentheorie* en *Berekenbaarheid en complexiteit* waren altijd meer dan interessant genoeg om vroeg voor op te staan. Mijn keuze voor een masterthesis in de grafentheorie met Gunnar als promotor was dan ook vlug gemaakt. Het was Kris die me erop wees dat ik eigenlijk al een masterdiploma had, en niet moest wachten op dat tweede diploma om voor een assistentenpositie te solliciteren. Zo belandde ik voor een jaar in de situatie dat ik tegelijkertijd assistent en student was.

Ongeveer gelijktijdig met mijn aanstelling als assistent (om exact te zijn drie dagen later) ontdekte ik vanuit mijn kot Kevin aan de overkant van de straat. Het appartement dat hij deelde met Jonas, zijn flatmate, was comfortabeler dan mijn kot. Na enkele bezoeken ben ik er niet meer weggegaan. Enkele jaren later kwam Ira er ook bij, waardoor we een soort gezin werden. Jonas en Ira, bedankt om het beste appartement van Gent met ons te delen. Ondertussen zijn jullie naar Londen verhuisd, maar altijd welkom om terug even in ons appartement te komen wonen als wij op reis zijn. Bedankt ook aan Erika, Rita, Mélanie en Denis om als buren het dorpsgevoel in de stad te creëren, en aan Eline en Edgar voor het voorbijlopen in de tuin en het komen piepen naast de haag.

Het appartement is natuurlijk niet het enige goede dat Kevin in mijn leven heeft gebracht. Ook de vele reizen, nieuwe vrienden, en als hint bedoelde kookboeken heb ik aan hem te danken. Kevin, bedankt voor je impulsieve ideeën en onvoorspelbaarheid die ervoor hebben gezorgd dat de afgelopen zes jaar nooit saai zijn geweest. Dankzij Kevin en zijn ouders versta ik ondertussen ook redelijk wat West-Vlaams. Gratiella en Patrick, bedankt voor de altijd gastvrije ontvangst in Ieper, de bezoeken aan Gent, de hulp bij het verhuizen en kuisen van het appartement, en de gezamenlijke reizen.

Dan zijn we nu bij het academische onderdeel aanbeland. Gunnar wil ik bedanken voor het onderwerp van mijn doctoraat, en de vier jaar als begeleider van mijn onderzoek met Nico als copromotor. Voor de laatste twee jaar, waarin ik vooral artikels en deze thesis heb geschreven, bedank ik Kris en Nico voor het nalezen, de kritische opmerkingen en suggesties, en de flexibele deadlines. Ook bedankt aan de leden van de jury: Carol, Jan, Jan, Tom en Tomo. Dankzij hun opmerkingen is vooral Hoofdstuk 2 nog veel verbeterd.

Mijn bureaugenoten door de jaren heen — Machteld, Jens, Koen, Oliver en Hans — wil ik bedanken om niet te raar op te kijken als ik op vreemde uren aankwam en vertrok. Omdat de helft van mijn werk de afgelopen zes jaar uit onderwijs bestond, met regelmatig wisselende vakken, heb ik ook een reeks mensen te bedanken voor de samenwerking op dat vlak. Eerst en vooral de proffen van de vakken waarvan ik de oefeningenlessen gegeven heb: Stijn, Willy, Gunnar, Veerle, en in het bijzonder David die het de hele periode met mij als assistent heeft moeten stellen. Vervolgens zijn er de andere assistenten waarmee ik ofwel samen les gaf, of die dezelfde les gaven aan een andere groep: Bart, Charlotte, Domien, Elke en Nico. Natuurlijk waren er ook minder serieuze momenten, bijvoorbeeld de vrijdagmiddagen met de onderzoeksgroep in het Fenikshof en de resto. Bedankt Christophe, Dieter, Gunnar, Herman, Kris, Nico en Veerle voor de interessante gesprekken. Hetzelfde geldt voor de collega's in de koffiepauzes, als ik niet vergat te gaan tenminste. In de andere middagpauzes in de resto zou ik nooit geraakt zijn zonder achtereenvolgens Catherine, Benoit, Dieter en Charlotte, die elke dag rondgingen en maar heel sporadisch vergaten dat er in de traphal ook bureau's zijn. Voor de organisatie van de spelletjesavonden en het jaarlijkse TWIkend (dat

dit jaar helaas niet is kunnen doorgaan) bedank ik Catherine, Nico, Joyce, Dieter, Benoit, Felix en Charlotte.

Dan zijn er nog vrienden waar ik buiten het werk veel aan heb gehad. Bedankt Dieter, Joyce, Nico en Sybren voor de reizen en uitstappen, knutselsessies, verhuishulp, filmfestivalbezoeken, autovervoer, en gewoon om er te zijn.

Tot slot zijn er enkele producten en diensten die onmisbaar waren voor de totstandkoming van dit doctoraat. De ijsmachine, de pastamachine, het kookboek van Fuchsia Dunlop en fietskoeriers voor de voedselvoorziening, de internetinstallateurs voor informatie- en enterainmentvoorziening, en alle reviews en webshops op het internet voor de materiaalvoorziening. En ook een heel klein beetje bedankt aan het virus dat heel de wereld een lockdown heeft opgelegd zodat ik in afzondering aan mijn doctoraat kon schrijven.

Samenvatting

Symmetrische veelvlakken komen op veel plaatsen voor. We vinden ze onder meer terug in chemische en biologische structuren, maar ook de mens heeft een duidelijke voorliefde voor symmetrie in gebouwen, gebruiksvoorwerpen, en kunstwerken. Op pagina 16 zien we bijvoorbeeld een verzameling symmetrische veelvlakken die worden gebruikt als voetbal of dobbelsteen. We noemen deze veelvlakken *symmetrisch* omdat we ze kunnen draaien en spiegelen op verschillende manieren en terug hetzelfde resultaat krijgen — op de getallen van de dobbelsteen na — met vlakken, zijdes en hoeken op dezelfde plaats in de ruimte. Het is dan ook logisch dat er al sinds de oude Grieken interesse is voor deze symmetrische structuren en hoe we ze kunnen construeren, en dat tot vandaag veel wiskundigen zich daar nog steeds mee bezighouden.

In Hoofdstuk 1 geven we een historisch overzicht van technieken om nieuwe symmetrische veelvlakken te construeren door "operaties" toe te passen op bestaande symmetrische veelvlakken. Deze methode is zo voor de hand liggend dat sommige van deze operaties door de geschiedenis heen meerdere keren werden ontdekt. Tegelijk blijkt uit dit overzicht dat er nog iets ontbreekt. We hebben wel een lijst van operaties die we kunnen toepassen op veelvlakken, maar er is geen overkoepelende theorie. Wanneer een operatie wordt omschreven, gebeurt dat dikwijls door "hoeken af te snijden", "zijdes op te delen", en andere ambachtelijke handelingen waar een wiskundige zich al snel ongemakkelijk bij voelt. Dit roept enkel vragen op. Hoe definiëren we een operatie? Hoe passen we ze toe? Wat maakt dat een operatie de symmetrie van een veelvlak bewaart? Hoeveel van deze operaties

zijn er? Om deze vragen te kunnen beantwoorden, hebben we een betere wiskundige omschrijving nodig van dit soort operaties.

Voor we daaraan kunnen beginnen, introduceren we in Hoofdstuk 2 enkele wiskundige concepten die we in latere hoofdstukken nodig zullen hebben. In deze thesis benaderen we de problemen *combinatorisch*. Voor ons is een veelvlak dus gewoon een *driesamenhangende vlakke graaf*, maar we zullen onze definities iets algemener maken zodat ze werken voor elke *ingebedde graaf*. Het is echter niet onmiddellijk duidelijk waarom deze combinatorische aanpak werkt, dus we beginnen met een korte introductie van de *topologische* aanpak, en tonen aan dat die equivalent is. De combinatorische aanpak zal het een stuk eenvoudiger maken om operaties te definiëren.

In Hoofdstuk 3 voeren we *lokale symmetriebewarende operaties* in. Dit zijn combinatorische operaties die op elke ingebedde graaf kunnen worden toegepast. Deze operaties zijn *lokaal* gedefinieerd, wat betekent dat we elk vlak, elke zijde en elke hoek van een veelvlak op dezelfde manier zullen transformeren. We bewijzen dat deze operaties effectief de symmetrie van een ingebedde graaf bewaren, en toegepast op een veelvlak altijd opnieuw een veelvlak zullen geven. Bovendien kunnen we operaties *samenstellen* en de mate waarin een operatie de grootte van een veelvlak doet toenemen (de *inflatiefactor*) berekenen. We kunnen dus "rekenen" met operaties. Tot slot tonen we aan dat een groot deel van de operaties beschreven in Hoofdstuk 1 tot de verzameling van lokale symmetriebewarende operaties behoren.

Nu we een formele definitie van lokale symmetriebewarende operaties hebben, is een voor de hand liggende vraag hoeveel van deze operaties er nog zijn buiten onze gekende voorbeelden. Daarom ontwikkelen we in Hoofdstuk 4 een *algoritme* om alle lokale symmetriebewarende operaties met een gegeven inflatiefactor te construeren. Dit algoritme zal vertrekken van enkele basisoperaties en deze op alle mogelijke manieren proberen uit te breiden tot operaties met een grotere inflatiefactor. Daarbij is het belangrijk om te vermijden dat we *isomorfe* operaties meerdere keren construeren. We willen namelijk zo weinig mogelijk tijd besteden aan het zoeken naar operaties die we al hadden gevonden, of aan uitbreidingen die uiteindelijk toch geen geldige operaties opleveren. Daarvoor gebruiken we enkele gekende technieken, zoals

de *canonieke constructiepadmethode* en het *homomorfismeprincipe*. Dit algoritme werd geïmplementeerd in een computerprogramma, waardoor we een lijst kunnen geven van alle lokale symmetriebewarende operaties met een gegeven inflatiefactor, en zien welke daarvan nog niet gekend zijn.

Zoals al eerder vermeld, zijn niet alle operaties die in Hoofdstuk 1 werden beschreven lokale symmetriebewarende operaties volgens onze definitie in Hoofdstuk 3. Toch zouden we ook deze operaties graag in ons theoretisch kader inpassen. Daarom breiden we in Hoofdstuk 5 onze definitie uit naar *lokale oriëntatiebewarendesymmetriebewarende operaties*. Deze operaties bewaren niet noodzakelijk alle symmetrie, maar ze bewaren wel de *oriëntatiebewarende* symmetrie. Dit worden ook wel *chirale* operaties genoemd. Net zoals in Hoofdstuk 3 bewijzen we dat deze operaties goed gedefinieerd zijn, effectief oriëntatiebewarende symmetrie bewaren en samengesteld kunnen worden tot nieuwe lokale oriëntatiebewarendesymmetriebewarende operaties. Voor deze operaties is het niet eenvoudig om te zien wanneer twee operaties eigenlijk dezelfde zijn. Daarom introduceren we *dubbelkamerdecoraties*, die ons toelaten om efficiënt te controleren of twee operaties *equivalent* zijn. Tot slot tonen we aan dat de resterende operaties uit Hoofdstuk 1 tot de verzameling van lokale oriëntatiebewarendesymmetriebewarende operaties behoren.

Vervolgens willen we ook alle chirale operaties met een gegeven inflatiefactor kunnen construeren. Het algoritme uit Hoofdstuk 4 is hiervoor echter niet bruikbaar. We ontwikkelen in Hoofdstuk 6 een volledig nieuw algoritme dat alle lokale oriëntatiebewarendesymmetriebewarende operaties construeert. Dit algoritme is voorlopig nog niet geïmplementeerd in een computerprogramma, maar we kunnen de technieken al wel gebruiken om met de hand een volledige lijst van chirale operaties (met kleine inflatiefactor) op te stellen.

Tot slot geven we in Hoofdstuk 7 een overzicht van onopgeloste problemen en vragen die interessant zijn om verder te onderzoeken.

Soccer Ball © Aaron Rotenberg / Wikimedia Commons / CC-BY-SA-4.0
Role Playing Dice © Diacritica / Wikimedia Commons / CC-BY-SA-3.0

Summary

Symmetric polyhedra are almost everywhere. We encounter them in nature and in chemical and biological structures, but humankind too has a clear preference for symmetry in buildings, objects and artworks. On page 16, we see a collection of symmetric polyhedra that are used as footballs or dice. We call these polyhedra *symmetric* because we can reflect and rotate them in different ways and get the same result — except for the numbers on the dice — with faces, edges and corners in the same locations. Thus, it is not surprising that there has been interest in these symmetric structures and how to construct them since the ancient Greeks, and that many mathematicians are still working on this today.

In Chapter 1, we give a historical overview of techniques to construct new symmetric polyhedra by applying "operations" to existing symmetric polyhedra. This method is so evident that some of these operations have been discovered multiple times throughout history. However, this overview also shows that something is still missing. We have a list of operations that we can apply to polyhedra, but there is no general framework. When an operation is described, it often happens with terminology like "cutting corners", "raising faces" and other DIY techniques that are rather unsettling for a mathematician. So this raises some questions. How should we define an operation? How do we apply them? What makes an operation preserve the symmetry of a polyhedron? How many such operations are there? To answer these questions, a better mathematical description for this kind of operations is required.

Before we get started, we introduce a few mathematical concepts in Chapter 2 that will be useful in later chapters. In this thesis we approach the problems in a *combinatorial* way. So we consider a polyhedron equivalent to a *three-connected plane graph*, but we will make our definitions more general so that they work for every *embedded graph*. However, it is not immediately clear why this combinatorial approach works, so we start with a brief introduction to the topological approach, and show that it is equivalent. The combinatorial approach will make it a lot easier to define operations.

In Chapter 3 we introduce *local symmetry preserving (lsp) operations*. These are combinatorially defined operations that can be applied to any embedded graph. They are defined *locally*, which means that we will transform every face, edge and corner of a polyhedron in the same way. We prove that these operations indeed preserve the symmetry of an embedded graph, and when applied to a polyhedron always result in a polyhedron. Furthermore, we can *compose* operations and calculate the extent to which an operation increases the size of a polyhedron (the *inflation factor*). Finally, we show that many of the operations described in Chapter 1 belong to the set of local symmetry preserving operations.

Now that we have a formal definition of local symmetry preserving operations, an obvious question is how many of these operations there are outside of our known examples. Therefore, we develop an algorithm to construct all local symmetry preserving operations with a given inflation factor in Chapter 4. This algorithm takes a few base operations as starting point, and tries to extend them in every possible way to operations with a larger inflation factor. It is important to avoid constructing *isomorphic* operations multiple times. We want to spend as little time as possible looking for operations that have been constructed already, or extending structures that ultimately do not yield valid operations. In order to do that we use some well-known techniques, such as the *canonical construction path method* and the *homomorphism principle*. This algorithm is implemented in a computer program, which allows us to list all local symmetry preserving operations with a given inflation factor, and determine which of them were previously unknown.

As mentioned before, not all operations described in Chapter 1 are local symmetry preserving operations according to our definition in Chapter 3. Nevertheless, we would like to fit these operations too into our theoretical framework. Therefore, we expand our definition to *local orientation-preserving symmetry preserving operations* in Chapter 5. These operations do not necessarily preserve all symmetry, but they do preserve all *orientation-preserving* symmetry. We call these *chiral* operations. Just as in Chapter 3, we demonstrate that these operations are well defined, preserve orientation-preserving symmetry and can be composed into new local orientation-preserving symmetry preserving operations. It can be difficult to determine whether two chiral operations are actually the same. Therefore we introduce *double chamber decorations* that allow us to check whether two operations are *equivalent*. Finally, we show that the remaining operations from Chapter 1 belong to the set of local orientation-preserving symmetry preserving operations.

Next, we also want to construct all chiral operations with a given inflation factor. However, the algorithm from Chapter 4 is not suitable for this problem. In Chapter 6, we develop a new algorithm that constructs all local orientation-preserving symmetry preserving operations. This algorithm has not yet been implemented in a computer program, but we can already use the techniques to manually compile a complete list of chiral operations (with small inflation factor).

Finally, in Chapter 7, we provide an overview of open problems and questions that require further research.

List of publications

G. Brinkmann, P. Goetschalckx and S. Schein. “Comparing the constructions of Goldberg, Fuller, Caspar, Klug and Coxeter, and a general approach to local symmetry-preserving operations”. In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 473.2206 (2017).

P. Goetschalckx, K. Coolsaet and N. Van Cleemput. “Generation of Local Symmetry-Preserving Operations”. In: *Ars Mathematica Contemporanea* (2020). eprint: 1908.11622.

P. Goetschalckx, K. Coolsaet and N. Van Cleemput. “Local Orientation-Preserving Symmetry Preserving Operations on Polyhedra”. In: *Discrete Mathematics* (2020). eprint: 2004.05501.

1

A brief history of operations on polyhedra

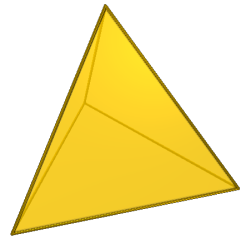
In this chapter, we will sketch the history of the construction of symmetric (convex) polyhedra. A common way of constructing new symmetric polyhedra is by applying some kind of operation to already existing polyhedra, without losing their symmetries. Later, we will try to fit all these operations into one mathematical framework.

We will give a better definition of convex polyhedra in Chapter 2, but for now a polyhedron is a 3-dimensional solid with flat polygonal faces, straight edges and vertices that form non-flat corners. A polyhedron is called *convex* if it is a convex subset of $\mathbb{R}^3$, i.e. it contains the whole line segment between any two points of the polyhedron.

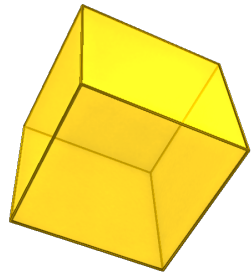
1 Platonic, Archimedean and Catalan solids

Before we can start talking about operations, we need some polyhedra to apply them to. We start with the most well-known and oldest ones.

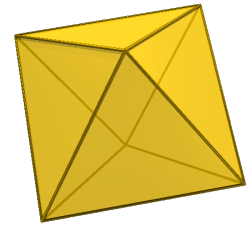
Definition 1.1. A *Platonic solid* is a convex polyhedron with regular faces. Each face has the same number of sides, and each vertex has the same degree.



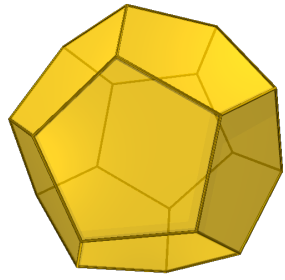
tetrahedron



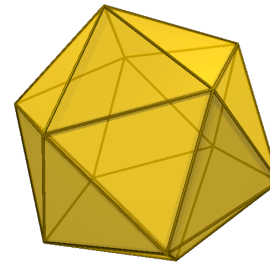
cube



octahedron



dodecahedron



icosahedron

Figure 1.1. The five Platonic solids.

The Platonic solids are vertex-transitive, edge-transitive and face-transitive, which means that their symmetry groups act transitively[1] on their vertices, edges and faces.

The ancient Greeks knew already that there exist only five Platonic solids. Some sources suggest that Plato (427–347 BC) only knew the cube, tetrahedron and dodecahedron. Theaetetus (417–369 BC) described all five of them, and was probably the first to prove that no other Platonic solids exist.

A few years later, Archimedes (287–212 BC) listed 13 other solids, which these days are called the *Archimedean solids*. His original work is lost, so we do not know for sure how these solids were constructed, but it is plausible that he applied some kind of operation to the Platonic solids.

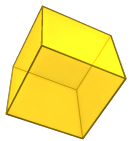
The first known written mention of the Archimedean solids was by Pappus of Alexandria (290–350 AD) in his book *Synagoge* [Eec33].

Definition 1.2. The *Archimedean solids* are convex vertex-transitive polyhedra with regular faces, excluding the Platonic solids, the prisms and antiprisms.

In 1620, Johannes Kepler (1571–1630) rediscovered the Archimedean solids and introduced his naming convention in his book *Harmonices Mundi* [Kep19]. In Table 1.1 the 13 Archimedean solids are displayed, together with Kepler's names. These names are derived from the operations that he used to construct the Archimedean solids from the Platonic solids.

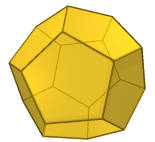
[1] for any two vertices v_1 and v_2 there is an automorphism f such that $f(v_1) = v_2$

T
tetrahedron



C
cube

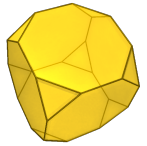
O
octahedron



D
dodecahedron

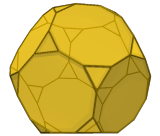
I
icosahedron

tT
truncated
tetrahedron

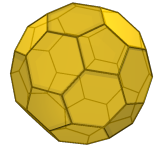


tC
truncated
cube

tO
truncated
octahedron

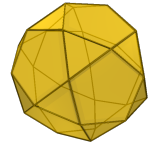


tD
truncated
dodecahedron

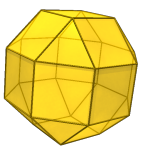


tI
truncated
icosahedron

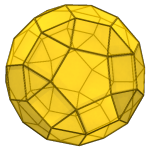
$aC = aO$
cuboctahedron



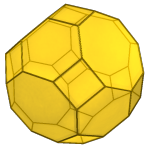
$aD = aI$
icosidodecahedron



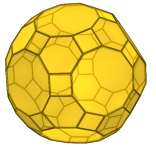
$eC = aaC = aaO$
rhombicuboctahedron



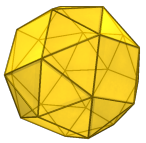
$eD = aaD = aaI$
rhombicosidodecahedron

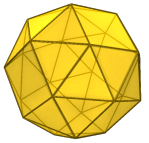


$bC = taC = taO$
truncated cuboctahedron

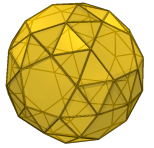


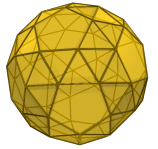
$bD = taD = taI$
truncated icosidodecahedron





$sC = sO$
snub cube



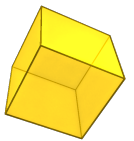


$sD = sI$
snub dodecahedron

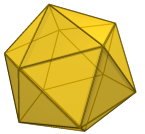
Table 1.1. The Archimedean solids.

$T = dT$
tetrahedron

$O = dC$
octahedron



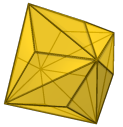
$C = dO$
cube



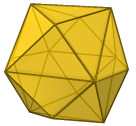
$I = dD$
icosahedron

$D = dI$
dodecahedron

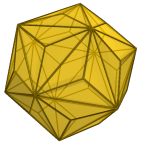
kT
triakis
tetrahedron



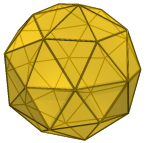
kO
triakis
octahedron



kC
tetrakis
hexahedron

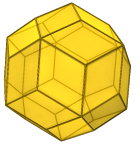


kI
triakis
icosahedron

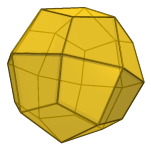


kD
pentakis
dodecahedron

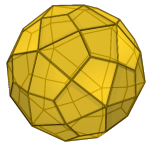
$jC = jO$
rhombic dodecahedron



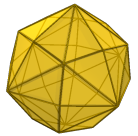
$jD = jI$
rhombic triacontahedron



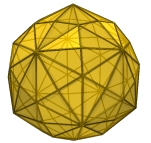
oC
deltoidal icositetrahedron



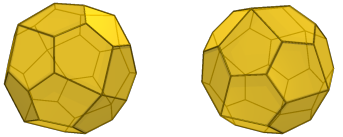
oD
deltoidal hexecontahedron



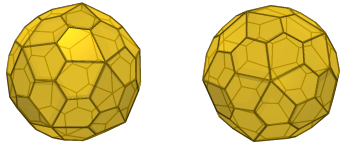
mC
disdyakis dodecahedron



mD
disdyakis triacontahedron



gC
pentagonal icositetrahedron



gD
pentagonal hexecontahedron

Table 1.2. The Catalan solids.

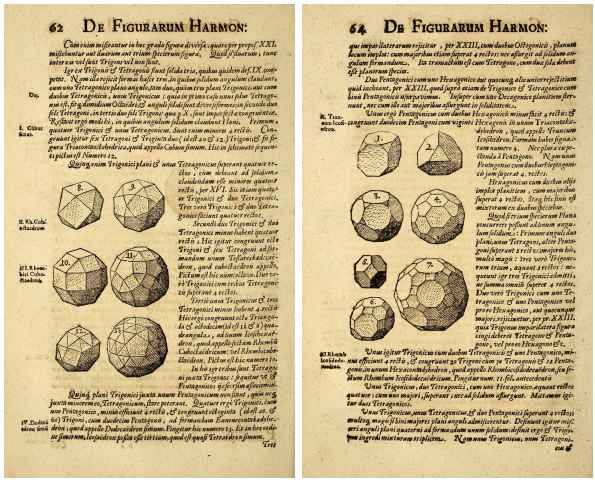


Figure 1.2. Pages from Kepler's *Harmonices Mundi*.

Remark 1.3. There are two versions of the snub cube and the snub dodecahedron in Table 1.1. The second one is the mirror image of the first. We will see later that snub is a *chiral* operation that can break mirror symmetry.

In 1965, the Belgian mathematician Eugène Catalan described the *Catalan solids* [Cat65], which are the duals of the Archimedean solids. As a consequence, the faces are no longer regular, but they are identical.

Definition 1.4. The *Catalan solids* are convex face-transitive[2] polyhedra with vertices of the same degree, excluding the Platonic solids, the bipyramids and trapezohedra.

The 13 Catalan solids are given in Table 1.2. Each polyhedron in this table is the dual of the corresponding Archimedean solid in Table 1.1.

In 1966, Norman Johnson [Joh66] published a list of 92 other convex polyhedra with regular faces, but without restrictions on the vertices. Victor Zalgaller [Zal67] proved in 1969 that this list is complete. We call these polyhedra the Johnson solids.

[2]This means that the automorphism group acts transitively on the faces.

2 Conway polyhedron notation

John Conway [CBG08] generalized the idea of constructing polyhedra with the same symmetry by applying operations to smaller ones. He introduced a new notation and defined a number of basic operations. Starting from a *seed polyhedron*, a sequence of operations can be applied. In Tables 1.1 and 1.2, the necessary operations to construct the Archimedean and Catalan solids are given. The Conway operations are listed in Table 1.3.

symbol	name	description
1	identity	do nothing
d	dual	swap vertices and faces
a	ambo	truncate until faces touch
j	join	add pyramids on faces, with joined sides
t	truncate	truncate all vertices
k	kis	add pyramids on faces
o	ortho	divide n-gons into n quadrangles
e	expand	replace vertices and edges by faces
s	snub	replace vertices by faces, edges by 2 triangles
g	gyro	divide n-gons into n pentagons
b	bevel	expand and replace n-gons by $2n$-gons
m	meta	divide n-gons into $2n$ triangles

Table 1.3. The basic Conway operations.

The descriptions of the Conway operations are ad hoc and rather informal. Some operations are equal to the composition of other operations, as can be seen in Tables 1.1 and 1.2. There are other relations that are not shown in these figures, and there are compositions of operations that have no name.

Remark 1.5. It is clear from the figures that there is something special about ambo. If we apply it to a pair of mutually dual polyhedra (e.g. a cube and an octahedron), the result is the same (a cuboctahedron), i.e. $a = ad$. Even more striking is that it seems to be essentially the only operation that can *increase* the symmetry. If we apply ambo to a self-dual polyhedron (e.g. a tetrahedron), the result (an octahedron) has more symmetry.

Later, more operations were added to the list — e.g. by George W. Hart [Har98]. Some of these can be expressed as the composition of other operations, and others are new. Hence, the list appears to be a zoo of operations, not a systematic classification. It is possible to come up with new operations that are missing from the list, but without a formal definition of 'operation', we cannot really speak of 'missing'.

3 The Goldberg-Coxeter construction

Another well-known — but at first sight unrelated — class of polyhedra are the *Goldberg polyhedra*. These are polyhedra with icosahedral symmetry, where all faces are hexagons or pentagons, and all vertices have degree 3. They have applications in organic chemistry, where they occur as *fullerenes* [DD04] with as the most famous example *buckminsterfullerene* [Kro+85]. Variations of this construction were independently discovered by Goldberg and Caspar and Klug, as pointed out by Coxeter, which has created some confusion that resulted in the name *Goldberg-Coxeter construction*.

A related class of polyhedra are *geodesic domes*, which have applications in architecture [MF73] and biology, where they e.g. occur as viral capsids [CK62].

3.1 Goldberg

In 1937, Michael Goldberg published his article "A Class of Multi-Symmetric Polyhedra" [Gol37]. In order to enumerate all 3-regular polyhedra with only pentagonal and hexagonal faces, he observes that each of these polyhedra can be obtained as a subdivision of the dodecahedron, with a pentagonal face in the center of each face of the dodecahedron, surrounded by hexagons.

Due to the required symmetry, each vertex of the original dodecahedron corresponds to a vertex or center of a hexagonal face of the new polyhedron, and each center of an edge of the dodecahedron corresponds to a vertex, center of an edge or center of a hexagonal face of the new polyhedron. Furthermore, the new polyhedron has to preserve the fivefold rotational symmetry in the centers of the faces of the dodecahedron.

So we can replace each face of the dodecahedron with five copies of the same triangular patch. This patch can be cut out of a hexagonal tiling as in Figure 1.3. If we can enumerate all these triangular patches, we can construct all required polyhedra.

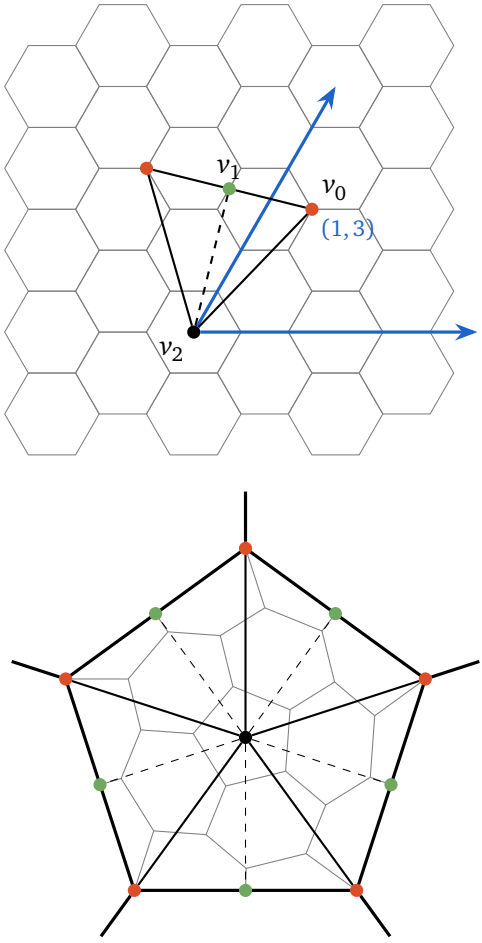


Figure 1.3. The $(1,3)$ Goldberg construction. An equilateral triangular patch determined by the point v_0 with coordinates $(1,3)$ is cut out of the hexagonal tiling. Five copies of this patch are pasted on a pentagonal face of a dodecahedron.

Lemma 1.6. *Assume the hexagonal tiling is equipped with a coordinate system such that* $(0,0)$ *is the center of a hexagonal face,* $(1,0)$ *is a vertex of this face, and* $(0,1)$ *is the vertex obtained by rotating* $(1,0)$ *by 60° in counterclockwise direction around* $(0,0)$.

A point (a,b) *is*

1. *the center of a face if and only if* a *and* b *are integers and* $a \equiv b \pmod 3$*;*
2. *a vertex if and only if* a *and* b *are integers and* $a \not\equiv b \pmod 3$*;*
3. *the center of an edge if and only if* a *and* b *are not both integers, but* $2a$ *and* $2b$ *are integers and* $2a \equiv 2b \pmod 3$.

Proof. We will first prove that a translation by (a,b) is an isomorphism of the hexagonal tiling if and only if a and b are integers and $a \equiv b \pmod 3$.

It is easy to see that each isomorphic translation (a,b) is the composition of translations by $(1,1)$ and $(-1,2)$. It follows immediately that $a \equiv b \pmod 3$. Now, consider an arbitrary translation (a,b) with $a \equiv b \pmod 3$. The result is isomorphic to a translation by $(a',b') = (0, b-a+3t)$ with $0 \le b' < 3$ since $(1,1)$ and $(0,3)$ are isomorphic translations. But $b' \equiv b-a+3t \equiv 0 \pmod 3$, and therefore $b' = 0$. It follows that (a',b') is the identity, and thus (a,b) is an isomorphic translation.

A point (a,b) is the center of a face if and only if it is the image of $(0,0)$ by an isomorphic translation, i.e. a and b are integers and $a \equiv b \pmod 3$.

A point (a,b) is a vertex if and only if it is the image of $(1,0)$ or $(2,0)$ by an isomorphic translation, i.e. a and b are integers and $a \not\equiv b \pmod 3$.

If (a,b) is the center of an edge, it is the image of

$$(a',b') \in \left\{\left(0,\frac{3}{2}\right),\left(\frac{1}{2},\frac{1}{2}\right),\left(\frac{1}{2},2\right)\right\}$$

by an isomorphic translation (x,y), i.e. $(a,b) = (a'+x, b'+y)$. Therefore, a and b are not both integers, but $2a$ and $2b$ are. Since

$2a' \equiv 2b' \pmod 3$, it follows that

$$2a \equiv 2(a'+x) \equiv 2a' \equiv 2b' \equiv 2(b'+y) \equiv 2b \pmod 3.$$

Suppose that a and b are not both integers, but $2a$ and $2b$ are integers and $2a \equiv 2b \pmod 3$. If a is an integer, we can translate the point (a,b) to $(a',b') = (0, b-a+3t)$ such that $0 \leq b' < 3$. Since $2b' \equiv 2(b-a+3t) \equiv 0 \pmod 3$ and b is not an integer, $b' = \frac{3}{2}$. Therefore, (a,b) and (a',b') are centers of edges. If a is not an integer, we can translate the point (a,b) to $(a',b') = \left(\frac{3}{2}, b-a+\frac{1}{2}+3t\right)$ such that $0 \leq b' < 3$. Since $2b' \equiv 2\left(b-a+\frac{1}{2}+3t\right) \equiv 1 \pmod 3$, b' is either $\frac{1}{2}$ or 2. Therefore, (a,b) and (a',b') are centers of edges. □

If we name the corners of the triangular patch v_0, v_1, v_2 as in Figure 1.3, and choose coordinates such that v_2 — which has to be the center of a face — is $(0,0)$, we can choose a vertex or center of a hexagon (a,b) for the position of v_0, and this uniquely determines $v_1 = \left(\frac{a-b}{2}, \frac{a+2b}{2}\right)$. It follows from Lemma 1.6 that the valid coordinates (a,b) are exactly the integer coordinates, and v_1 is the center of a face or edge. By only using coordinates with $a > 0$ and $b \geq 0$, we ensure that v_0 is always chosen in the same 60° sector, avoiding congruent patches. The parameters (a,b) are used to identify each Goldberg operation.

It is a direct consequence of the symmetry of the hexagonal tiling that the glued together edges of two neighbouring patches nicely match. Since the vertices of the dodecahedron have degree 3, the local structure around these vertices is the same as in the hexagonal tiling. Only in the centers of the faces of the dodecahedron, the structure is different, and a pentagon is formed. All other faces are still hexagons, and all vertices have degree 3.

The example in Figure 1.3 results in a polyhedron with chiral symmetry. In Figure 1.4, an achiral example is given. Each patch consists of a pair of two mirrored triangular patches. This situation occurs when v_2 and v_1 lie on a mirror axis of the hexagonal tiling, i.e. v_0 is of the form $(a,0)$ or (a,a) for $a > 0$.

Instead of cutting out the larger patch, we can cut out this smaller patch and glue it together with its mirror image on the faces of a

dodecahedron. In fact, this is the approach that Goldberg described in [Gol37]:

> Each pentagonal patch may be divided into ten equivalent (congruent or symmetric) triangular patches.

This is true in the achiral case (see Figure 1.4), but obviously not in the chiral case (see Figure 1.3), for which the ten triangular patches are divided between two non-equivalent classes. This mistake is surprising, given that he included a picture of the $(5,3)$ patch.

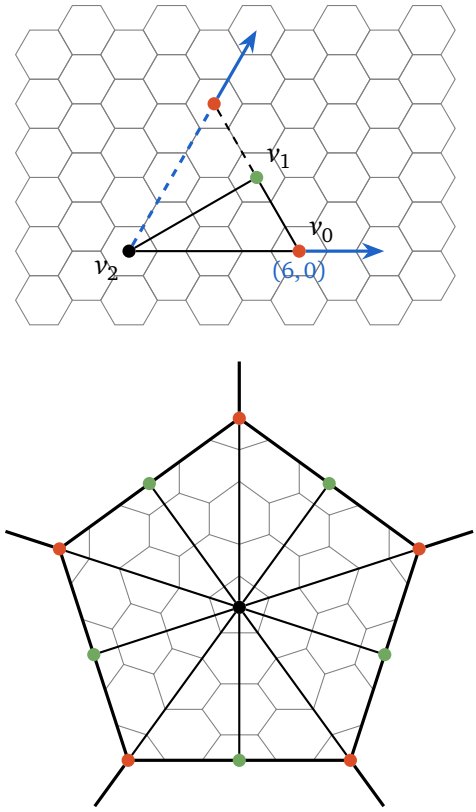


Figure 1.4. The $(6,0)$ Goldberg construction. The equilateral triangular patch consists of two equivalent right-angled triangular patches.

3.2 Buckminster Fuller

Between 1948 and 1959, the architect Buckminster Fuller designed and constructed what he called *geodesic domes*, for which he received a U.S. patent in 1954. A geodesic dome is a spherical structure consisting of equilateral triangles. Because of the structural rigidity of the triangular elements, a geodesic dome can withstand heavy loads and weather, while staying lightweight and relatively cheap itself. A downside is that less off-the-shelf materials can be used, which increases the cost.

Buckminster Fuller was not the original inventor of geodesic structures. In 1926, a geodesic dome was designed by Walther Bauersfeld for the Zeiss-Planetarium in Jena, Germany (see Figure 1.5). It was based on a subdivided icosahedron in order to get close to a round surface for the projection of the stars.

Buckminster Fuller popularized the idea in the United States, which is why today it is mainly his name that is associated with geodesic domes. Some examples of his geodesic domes are given in Figures 1.6 and 1.7.

Most geodesic domes are subdivided icosahedra. Buckminster Fuller's *Dymaxion world map*, which is the projection of a sphere on an inscribed polyhedron that can be unfolded and flattened, was first based on a cuboctahedron (which he called *vector equilibrium*) and in later versions on an icosahedron. We are of course primarily interested in how these structures are conceived. In [MF73], Buckminster Fuller's friend Robert Marks writes:

> The geometric principles underlying the Dymaxion map are the same as those used to develop the basic pattern of Fuller's domes.
>
> The domes are based on the fact that a regular geometric "solid" (such as an icosahedron) can be projected outwardly on to the surface of a sphere (thus an ordinary icosahedron generates a spherical icosahedron).
>
> To produce Fuller's Dymaxion map, we reverse this process. We start with a sphere, on whose surface a spherical icosahedron has been drawn. Next, we subtriangulate the

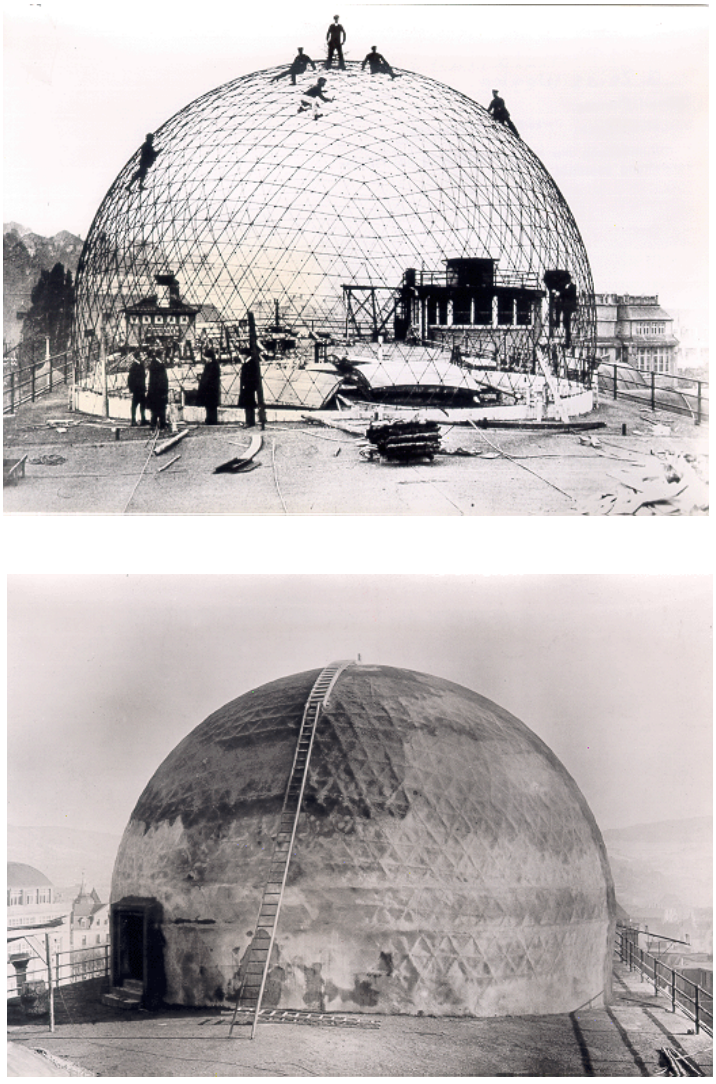


Figure 1.5. The Zeiss-Planetarium in Jena, Germany (1926).

Figure 1.6. The R. Buckminster Fuller and Anne Hewlett Dome Home in Carbondale, Illinois, USA (1960).

Figure 1.7. The Biosphere in Montreal, Quebec, Canada (1967).

icosahedron's 20 triangular faces with symmetric, three-way, great circle grids of a chosen frequency. Then we transfer this figure's configuration of points to the faces of an ordinary (non-spherical) icosahedron which has been symmetrically subtriangulated in frequency of modular subdivision corresponding to the frequency of the spherical icosahedron's subdivisions.

3.3 Caspar and Klug

In 1956, Watson and Crick speculated in [CW56] on the structure of small viruses. They suggested that a lot of small viruses consist of identical subunits, packed together in a regular manner. Since viruses are made of protein and ribonucleic acid, the number of ways they can form a spherical shell is limited. The only plausible candidates have tetrahedral, octahedral or icosahedral symmetry.

In later experimental studies, icosahedral symmetry appeared to be the most common, but the number of identical molecules was larger than expected.

Inspired by the resemblance between the geometric structure of Buckminster Fuller's geodesic domes and icosahedral viruses, Caspar and Klug [CK62] developed a way to enumerate possible methods of arranging the subunits of an icosahedral virus. Their approach is similar to Goldberg, but they work in the dual polyhedron. They cut out a equilateral triangle of a triangular tiling, and paste this patch on each face of an icosahedron, as illustrated in Figure 1.8. The result of this example is shown in Figure 1.9.

Assume the triangular tiling is equipped with a coordinate system such that $(0,0)$ is a vertex, $(1,0)$ is a neighbouring vertex, and $(0,1)$ is the vertex obtained by rotating $(1,0)$ by $60°$ in counterclockwise direction around $(0,0)$. If we name the corners of the triangular patch v_0, v_0', v_0'' as in Figure 1.8, and choose coordinates such that $v_0 = (0,0)$, we can choose a vertex (h,k) for the position of v_0', and this uniquely determines $v_0'' = (-k, h+k)$. The parameters (h,k) are used to identify each Caspar-Klug operation.

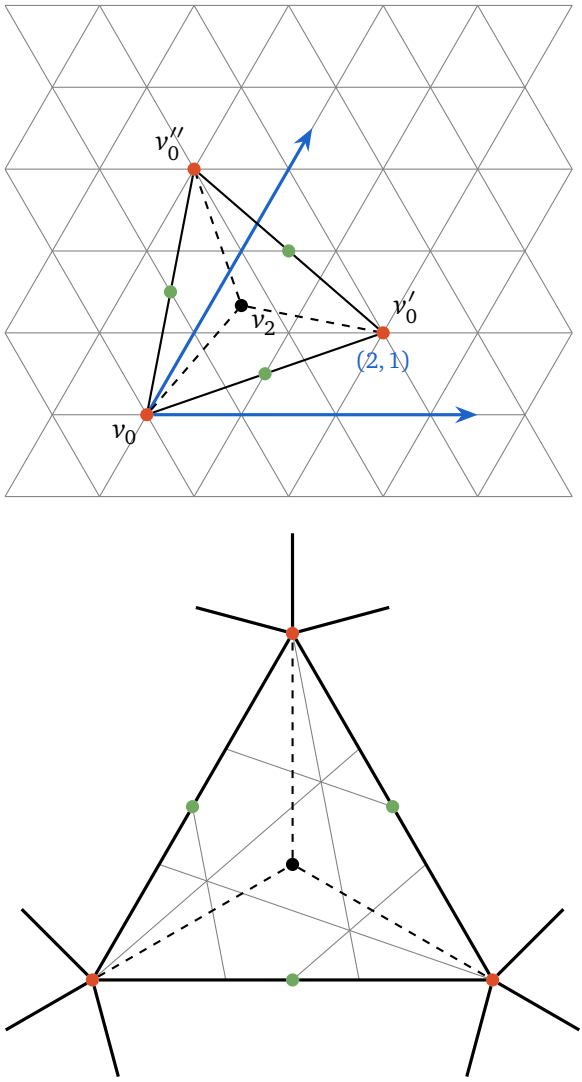


Figure 1.8. The $(2,1)$ Caspar-Klug construction. An equilateral triangular patch determined by the point v_0' with coordinates $(2,1)$ is cut out of the triangular tiling. A copy of this patch is pasted on each triangular face of an icosahedron.

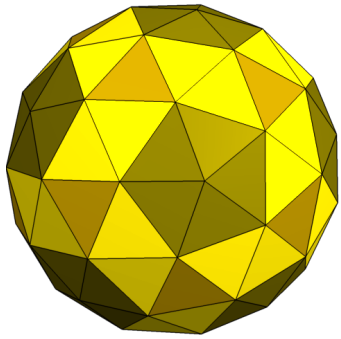


Figure 1.9. The icosadeltahedron constructed in Figure 1.8.

An important difference with the approach of Goldberg is that the construction of Caspar and Klug can only be applied to the icosahedron (or other polyhedra with triangular faces), since they cut out complete faces. Goldberg's construction is more general, cutting out smaller triangular patches to fill an n-gon with n copies of the same patch.

For a more detailed justification of their construction, Caspar and Klug referenced a paper that never got published, as Caspar comments in [Cas84]. Caspar and Klug developed this construction independently of Goldberg, but in their next publication [CK63] on the subject they mention Goldberg's work.

Theorem 1.7. *The polyhedron obtained by applying the Goldberg operation with parameters (a, b) to a dodecahedron is the dual of the polyhedron obtained by applying the Caspar-Klug operation with parameters (a, b) to an icosahedron.*

Proof. The dual of a hexagonal tiling is a triangular tiling. If we cut a Goldberg (a, b) patch out of the hexagonal tiling, and paste five copies of this patch on each pentagonal face of a dodecahedron as in Figure 1.10, it is clear that the dual of this polyhedron can be obtained by cutting the indicated triangular patch out of the corresponding triangular tiling and pasting this onto each triangular face of an icosahedron, which is the dual of a dodecahedron. We still have to prove that the parameters (h, k) are equal to (a, b).

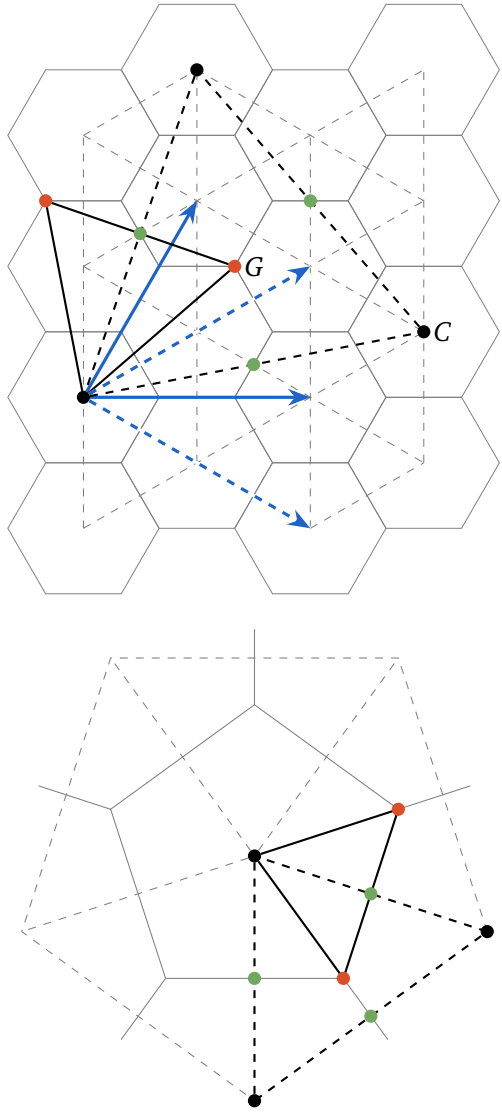


Figure 1.10. The $(1,2)$ Goldberg construction is drawn in solid lines, with the point G at coordinates $(1,2)$ relative to the solid axes. The $(1,2)$ Caspar-Klug construction is drawn in dashed lines, with the point C at coordinates $(1,2)$ relative to the dashed axes. The Goldberg patch is pasted on the pentagonal faces of a dodecahedron, while the Caspar-Klug patch is pasted on the triangular faces of the dual icosahedron.

Suppose A, B are the unit vectors of the coordinate system used by Goldberg, and H, K are the unit vectors of the coordinate system used by Caspar and Klug, both with the same origin. We obtain

$$A = \frac{H+K}{3} \quad \text{and} \quad B = \frac{2K-H}{3}.$$

The parameters (a, b) are the coordinates of the point G such that $G = aA + bB$. The parameters (h, k) are the coordinates of the point C such that $C = hH + kK$. Since C can be obtained as the sum of the vectors G and G rotated by 60°in clockwise direction, we have

$$\begin{aligned} C &= aA + bB + a(A-B) + bA \\ &= (2a+b)A + (b-a)B \\ &= (2a+b)\frac{H+K}{3} + (b-a)\frac{2K-H}{3} \\ &= aH + bK \end{aligned}$$

and it follows that $(a, b) = (h, k)$. □

3.4 Coxeter

In 1971, H.S.M. Coxeter published a survey article [Cox71] that refers to Goldberg and Caspar and Klug, and described the construction of the latter in detail. Subsequently, the literature often refers to the Goldberg-Coxeter construction, although this construction is actually that of Caspar and Klug.

In mathematics and chemistry, the articles of Goldberg [Gol37] and Coxeter [Cox71] are more popular than the ones by Caspar and Klug [CK62; CK63], which have a more biological focus and are vague in the mathematical details. This is probably the cause of the confusion between them. The sentence

> Independently of Michael Goldberg, Caspar and Klug proposed the following rule for making a suitable pattern.

in [Cox71] suggests that the construction of Goldberg is the same as the construction of Caspar and Klug, which is not true.

4 Other approaches

There are several other techniques to construct polyhedra not yet mentioned in this chapter. We will give a very brief overview.

The *Wythoff construction* [Wyt18; Cox73] is a method to construct *uniform polyhedra*, i.e. vertex-transitive polyhedra with regular faces. A sphere is divided in spherical *Schwarz triangles*, and one vertex is placed in each triangle. The *Wythoff symbol* determines the placement of the vertex and the angles of the triangles. So there is no explicit seed polyhedron, but the subdivided sphere can be considered a polyhedron.

Goldberg's idea of using operations on polyhedra to construct fullerenes was extended and further investigated in [FM06]. The *leapfrog* operation, which is a symmetry preserving operation on fullerenes, was generalized to other polyhedra in [FP94].

Another recent effort to formalize operations on oriented maps using *arc graphs* [PWB17], is dual to our approach in Chapter 3.

References

[Cas84] D.L.D. Caspar. “This Week’s Citation Classic”. In: *Current Contents Life Sciences* 4 (1984), p. 168.

[CK62] D.L.D. Caspar and A. Klug. “Physical Principles in the Construction of Regular Viruses”. In: *Cold Spring Harbor Symposia on Quantitative Biology*. Vol. 27. 1962, pp. 1–24.

[CK63] D.L.D. Caspar and A. Klug. “Structure and Assembly of Regular Virus Particles”. In: *Viruses, Nucleic Acids and Cancer. A Collection of Papers Presented at the 17th Annual Symposium on Fundamental Cancer Research*. The Williams and Wilkins Company, 1963, pp. 27–39.

[Cat65] M.E. Catalan. “Mémoire sur la théorie des polyèdres”. In: *Journal de l’école Impériale Polytechnique* XLI (1865), pp. 1–71.

[CBG08] J.H. Conway, H. Burgiel and C. Goodman-Strauss. *The Symmetries of Things*. A K Peters, 2008.

[Cox71] H.S.M. Coxeter. “Virus Macromolecules and Geodesic Domes”. In: *A Spectrum of Mathematics. essays presented to H.G. Forder*. Ed. by J.C. Butcher. Auckland University Press, 1971, pp. 98–107.

[Cox73] H.S.M. Coxeter. *Regular Polytopes*. 3rd ed. Dover Publications, 1973.

[CW56] F.H.C. Crick and J.D. Watson. “Structure of Small Viruses”. In: *Nature* 177 (1956), pp. 473–475.

[DD04] M. Dutour and M. Deza. “Goldberg–Coxeter Construction for 3- and 4-valent Plane Graphs”. In: *The Electronic Journal of Combinatorics* 11 (2004).

[Eec33] P. Ver Eecke. *Pappus d’Alexandrie: La Collection Mathématique*. Desclée de Brouwer, 1933.

[FM06] P.W. Fowler and D.E. Manolopoulos. *An Atlas of Fullerenes*. 2nd ed. Dover Publications, 2006.

[FP94] P.W. Fowler and T. Pisanski. “Leapfrog Transformations and Polyhedra of Clar Type”. In: vol. 90. 19. 1994, pp. 2865–2871.

[Gol37] M. Goldberg. “A Class of Multi-Symmetric Polyhedra”. In: *Tohoku Mathematical Journal*. First Series 43 (1937), pp. 104–108.

[Har98] G.W. Hart. *Conway Notation for Polyhedra*. 1998. URL: http://www.georgehart.com/virtual-polyhedra/conway_notation.html.

[Joh66] N.W. Johnson. “Convex Polyhedra with Regular Faces”. In: *Canadian Journal of Mathematics* 18 (1966), pp. 169–200.

[Kep19] J. Kepler. *Ioannis Keppleri Harmonices mundi libri V*. Linz, 1619.

[Kro+85] H.W. Kroto et al. “C_{60}: Buckminsterfullerene”. In: *Nature* 318 (1985), pp. 162–163.

[MF73] R.W. Marks and R. Buckminster Fuller. *The Dymaxion world of Buckminster Fuller*. Doubleday Anchor Books, 1973.

[PWB17] T. Pisanski, G. Williams and L.W. Berman. “Operations on Oriented Maps”. In: *Symmetry* 9 (2017), p. 274.

[Wyt18] W.A. Wythoff. “A relation between the polytopes of the C600-family”. In: *Proceedings of the Section of Sciences* 20 (1918), pp. 966–970.

[Zal67] V.A. Zalgaller. “Convex Polyhedra with Regular Faces”. In: *Zap. Nauchn. Semin. Leningr. Otd. Mat. Inst. Steklova* 2 (1967), pp. 1–221.

2

Embedded graphs, tilings and polyhedra

This chapter formally introduces some essential concept of graph theory. A more elaborate introduction can be found in [Die97] and [GYZ14].

There are different definitions for embedded graphs, tilings and polyhedra depending on the context. There is a combinatorial approach, that is used in the following chapters, and a topological approach. In this chapter, the relation between the two is explained.

1 Combinatorial graphs

In a combinatorial context, a graph is generally defined as follows:

Definition 2.1. A *graph* G consists of a vertex set V and an edge set E. Each edge has one or two vertices associated to it, which are called its *endpoints*.

We call a vertex v and an edge e *incident* to each other if v is an endpoint of e. Two vertices are called *adjacent* if there is an edge in E that connects these vertices.

The *degree* of a vertex is the number of incident edges. The total number of edges is the *size* of the graph, and the total number of vertices is the *order* of the graph.

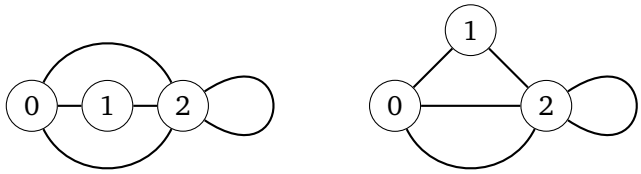


Figure 2.1. Two drawings of the same graph with one double edge and one loop on vertex set $V = \{0, 1, 2\}$.

This definition allows edges with only one endpoint — which we call *loops* — and multiple edges with the same endpoints — which we call *double edges*. For some applications, these are forbidden.

Definition 2.2. A *simple graph* is a graph that has no loops or double edges.

Sometimes it is useful to give a direction to the edges of a graph.

Definition 2.3. A *directed graph* or *digraph* is a graph with directed edges, i.e. one of the endpoints of each edge is designated as the *source*, and the other (or the same if there is only one) as the *target*.

The *underlying graph* of a directed graph is obtained by removing the source and target designations.

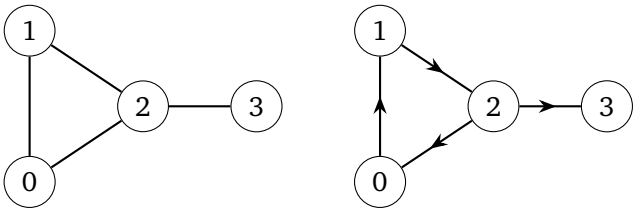


Figure 2.2. A simple graph and a directed graph with the same underlying graph.

A *path* between vertices v and w in a graph G is a sequence of edges $e_0, \ldots, e_n$ such that v is an endpoint of e_1, w is an endpoint of e_n, and e_i has a common endpoint v_i with e_{i+1} for every $0 \leq i < n$. A *simple path* is a path in which all vertices v_i are distinct. A *cycle* is a path from a vertex v to itself.

We say that a graph is *connected* if there exists a path between every pair of vertices. A graph is said to be *k-connected* if it has more than k vertices and remains connected if fewer than k vertices are removed. A vertex that makes a graph disconnected when removed is called a *cut-vertex*, and an edge that makes a graph disconnected when removed is called a *cut-edge* or *bridge*. The graph in Figure 2.1 is 2-connected but not 3-connected, since it becomes disconnected if vertices 0 and 2 are removed. The simple graph in Figure 2.2 is connected but not 2-connected, since it becomes disconnected if vertex 2 is removed.

2 Embedded graphs

In topological graph theory, graphs are drawn on surfaces. From the topological embedding of a graph, we can derive its combinatorial embedding, which gives an ordering to the edges incident to a vertex. In the following chapters, we will only use the combinatorial embedding.

Definition 2.4. The *topological representation* of a graph is obtained by identifying each edge with a copy of the closed interval $[0, 1]$, and identifying endpoints of intervals if the corresponding endpoints of the edges coincide.

Topological representations can be embedded in other topological spaces, but we will limit ourselves to *2-cell embeddings* on *orientable closed surfaces*. A closed surface is a compact, connected 2-manifold. A surface is orientable if we can define the concept of clockwise rotation in a continuous consistent matter, i.e. a directed loop can not be continuously deformed to a loop in the opposite direction, like on a Möbius band.

Definition 2.5. An *embedding* of a connected graph G on an orientable closed surface S is a continuous injective function from the topological representation of G into S. The connected components of the complement of the image are the *faces*. If each face is homeomorphic to an open disk, it is a *2-cell embedding*.

An example of a topological embedding that is not a 2-cell embedding is the embedding of a cycle graph on the torus.

A 2-cell embedding of a finite connected graph on a closed surface is sometimes called a *map*. A map can be described combinatorially by its *flag graph*, where a *flag* is a triple consisting of a mutually incident vertex, edge and face (see [GR01]). If the surface is orientable, there is an equivalent *rotation system*, which we call the combinatorial embedding.

Definition 2.6. A (combinatorially) *embedded graph* G is a triple (E, σ, θ) where E is a finite set, σ is a permutation of E and θ is a fixed-point-free involution of E.

The set E is the set of *directed edges* of G. We call a $\langle\sigma\rangle$-orbit a *vertex* of G, a $\langle\sigma^{-1}\theta\rangle$-orbit a *face* of G, and a $\langle\theta\rangle$-orbit an (undirected) *edge* of G. The degree of a vertex or the size of a face is the size of its corresponding orbit.

The corresponding combinatorial embedding of a topological embedding of a graph G can be obtained by taking E the set of directed edges of G, $\sigma(e)$ the next edge with the same source vertex as e in clockwise[1] direction, and $\theta(e)$ the inverse directed edge of e (see Figure 2.3).

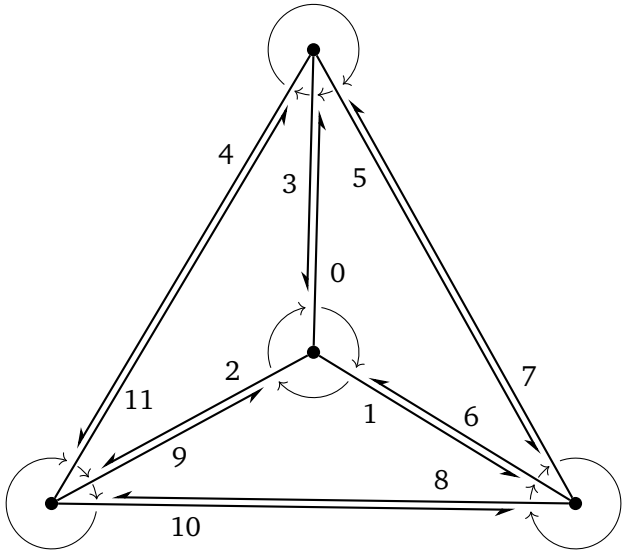
Given a combinatorially embedded graph, the induced topological 2-cell embedding can be obtained by "tracing the faces". Informally, we identify the combinatorial faces with open disks and identify their borders along the edges of the graph. This induced embedding is unique (see [GT01] for a proof).

By ignoring the ordering implied by σ — only considering the vertices and edges between them — we can reconstruct the underlying graph of an embedded graph.

Note that a face can contain both directions e and $\theta(e)$ of an edge. In that case, e is a cut-edge in the underlying graph. If a face contains the same vertex multiple times, this vertex is a cut-vertex in the underlying graph. Both of these situations occur in Figure 2.4.

An embedded graph with only faces of size 3 is called a *triangulation*, and an embedded graph with only faces of size 4 is called a *quadrangulation*.

[1] On an orientable surface, we can choose a clockwise direction.



$$E = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$$

$$\begin{aligned}
\sigma(0) &= 1 & \theta(0) &= 3 \\
\sigma(1) &= 2 & \theta(1) &= 6 \\
\sigma(2) &= 0 & \theta(2) &= 9 \\
\sigma(3) &= 4 & \theta(3) &= 0 \\
\sigma(4) &= 5 & \theta(4) &= 11 \\
\sigma(5) &= 3 & \theta(5) &= 7 \\
\sigma(6) &= 7 & \theta(6) &= 1 \\
\sigma(7) &= 8 & \theta(7) &= 5 \\
\sigma(8) &= 6 & \theta(8) &= 10 \\
\sigma(9) &= 10 & \theta(9) &= 2 \\
\sigma(10) &= 11 & \theta(10) &= 8 \\
\sigma(11) &= 9 & \theta(11) &= 4
\end{aligned}$$

Figure 2.3. An embedded graph $G = (E, \sigma, \theta)$ with 4 vertices of degree three, 12 directed edges, and 4 triangular faces.

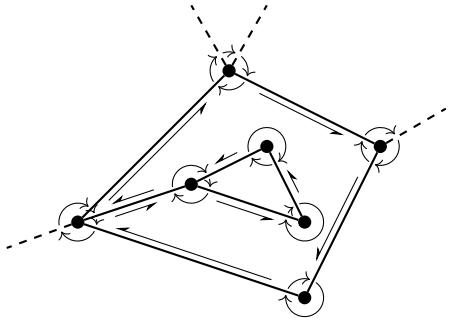


Figure 2.4. A face of size 9 in an embedded graph, containing a cut-edge and two cut-vertices.

Definition 2.7. The *dual graph* of an embedded graph $G = (E, \sigma, \theta)$ is the graph

$$G' = (E, \sigma^{-1}\theta, \theta).$$

It is clear from the definition of faces and vertices that the faces of the dual graph G' are the vertices of G and vice versa. Topologically, each edge has a dual edge that connects the centers of the faces on both sides (see Figure 2.5).

We are mainly interested in graphs that can be embedded in the 2-dimensional plane. The plane is not a closed surface, but since it differs from the sphere by only a single point, a graph can be embedded in the plane if and only if it can be embedded on the sphere. Some graphs can only be embedded on surfaces with higher genus. This property can be characterized combinatorially.

Definition 2.8. The *Euler characteristic* of an embedded graph $G = (E, \sigma, \theta)$ with vertex set V and face set F is

$$\chi(G) = |V| - |E| + |F|.$$

A *plane graph* is an embedded graph G with $\chi(G) = 2$. A *planar* graph is a graph G for which there exists a planar embedding (i.e. a plane graph with G as underlying graph).

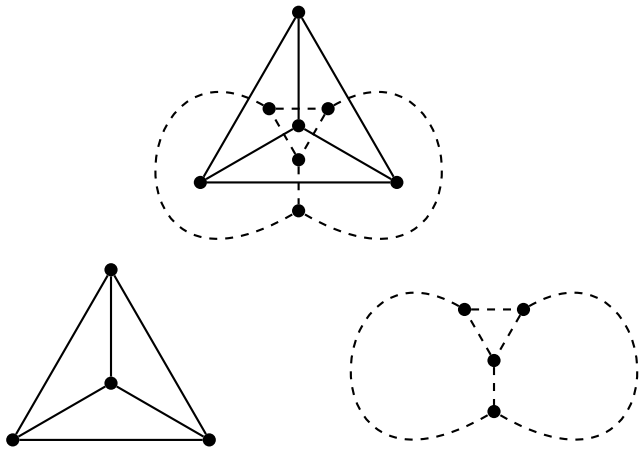


Figure 2.5. The embedded graph of the tetrahedron with its dual graph. In the first figure, we see the correspondence between the edges of a graph and the edges of its dual graph. In this example, the dual graph of the tetrahedron is again a tetrahedron.

3 Symmetry and isomorphisms

Two graphs G and G' are *isomorphic* if there exists a bijection f between their vertices such that (u, v) is an edge in G if and only if $(f(u), f(v))$ is an edge of G'. Since we can embed the same graph in different ways, we need a stronger definition for isomorphisms of embedded graphs.

Definition 2.9. An *orientation-preserving isomorphism* between two embedded graphs $G = (E, \sigma, \theta)$ and $G' = (E', \sigma', \theta')$ is a bijection $f : E \to E'$ such that

$$f(\sigma(e)) = \sigma'(f(e)) \quad \text{and} \quad f(\theta(e)) = \theta'(f(e)).$$

An *orientation-reversing isomorphism* is a bijection such that

$$f(\sigma(e)) = \sigma'^{-1}(f(e)) \quad \text{and} \quad f(\theta(e)) = \theta'(f(e)).$$

An *automorphism* is an isomorphism from an embedded graph to itself. We call two embedded graphs for which there exists an orientation-preserving or orientation-reversing isomorphism

orientation-preserving resp. *orientation-reversing isomorphic*. We call two embedded graphs *isomorphic* if they are orientation-preserving or orientation-reversing isomorphic. An embedded graph that is orientation-reversing isomorphic to itself is called *chiral*, and a graph that is not chiral is called *achiral*.

The *symmetry group* of an embedded graph G is the group of automorphisms of G.

Note that an orientation-preserving or orientation-reversing isomorphism between two embedded graphs implies a isomorphism between their underlying graphs.

An orientation-preserving or orientation-reversing isomorphism of a plane graph is completely determined by the image $f(e)$ of one directed edge e. The images of all other edges can be derived from this edge by applying the same sequence of operations σ and θ to e and $f(e)$.

As a consequence, we can easily derive a canonical form for plane graphs. Given a directed edge e_1, label the source vertex v_1 with 1, and the neighbours of v_1 starting with the target of e_1 with increasing numbers in clockwise order. Next, take the vertex v_2 with label 2 and label the unlabelled neighbours of v_2 with increasing numbers in clockwise order, starting from the already labelled neighbour with the lowest label. Do this for all vertices $v_1, \ldots, v_n$, until all vertices are labelled. We can now write this labelling down as a sequence of numbers by giving all labels of neighbours of v_i starting with the one with the lowest label, for i from 1 to n and separated by 0's. The choice of e_1 that results in the lexicographically lowest sequence is a *canonical starting edge*, and the corresponding sequence is the *canonical code* of the embedded graph. It is possible that there is more than one canonical starting edge, but they will belong to the same orbit. See Figure 2.6 for an example.

When actually computing and using canonical codes, there are some optimizations possible [BM07]. It is clear that the canonical starting edge will always start in a vertex with minimum degree, so it is not necessary to consider starting edges with higher degrees. This

optimization can be extended to other vertices by adding the degree of each vertex to the code (e.g. the first time a vertex occurs).

4 Tilings

Although we limited Definition 2.6 to finite graphs, we will need some infinite graph embeddings in later chapters. We start again with a topological definition [GS87].

Definition 2.10. A *topological tiling* is a countable family of closed sets or *tiles* which cover the Euclidean plane without gaps or overlaps.

This definition is far too general for our purposes. Therefore, we impose several restrictions.

Definition 2.11. A topological tiling is called *normal* if it satisfies the following three conditions:

1. every tile is a topological disk;
2. the intersection of two tiles is a connected set;
3. the tiles are uniformly bounded.

The first condition is similar to the conditions of a 2-cell embedding, with tiles corresponding to faces. The second condition ensures that the intersection of two tiles is either empty, a single point which we call a *vertex* of the tiling, or a one-dimensional arc which we call an *edge* of the tiling. The third condition means that there are two numbers l and u such that every tile contains a circular disk of radius l and is contained in a circular disk of radius u, i.e. there are no arbitrarily small or large tiles.

It follows from these properties that the vertices and edges of a normal tiling form an infinite combinatorial graph with minimum degree at least 3.

Definition 2.12. A topological tiling is called *periodic* if its symmetry group contains at least two translations in non-parallel directions.

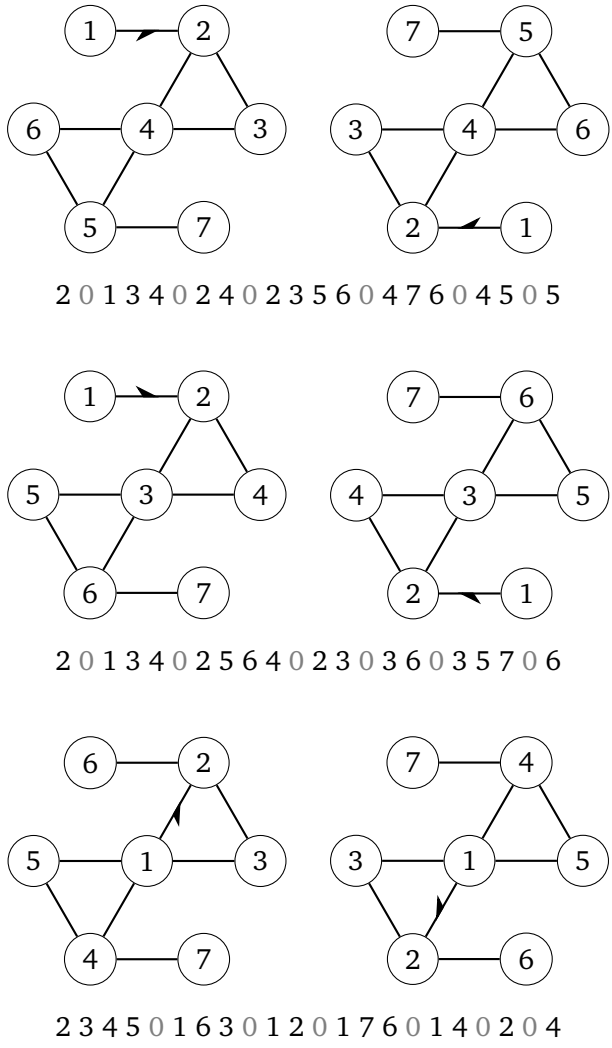


Figure 2.6. Different numberings of the same plane graph with corresponding code. The starting edge and orientation is designated by an arrow. Some starting edges result in the same code, since there is an automorphism that maps these edges onto each other. The first two numberings results in the lexicographically lowest canonical code.

The possible symmetry groups of periodic tilings are known as the *wallpaper groups* [GS87; CBG08].

The embedded infinite graph corresponding to a normal periodic tiling can be characterized combinatorially.

Definition 2.13. A *periodic tiling* is a plane graph T with vertex set V, edge set E and face set F infinite, but with finite maximum degree and face size, minimum degree at least 3, and a symmetry group G such that the number of orbits of vertices, edges and faces is finite.

Some examples that already appeared in Chapter 1 are the regular hexagonal and triangular tilings, which are each other's dual. A tiling is said to be regular if the symmetry group acts transitively on the flags of the tiling. The only other regular tiling is the square tiling.

5 Polyhedra

There are a lot of different approaches to define polyhedra. Some define a polyhedron as a solid, and others as a surface. Some allow self-crossing and others do not. And as Grünbaum said in [Grü94], often it is not defined at all.

> The Original Sin in the theory of polyhedra goes back to Euclid, and through Kepler, Poinsot, Cauchy and many others continues to afflict all the work on this topic. [...] at each stage [...] the writers failed to define what are the "polyhedra".

Although the term "convex polyhedron" suggest to be a special case of a general polyhedron, a convex polyhedron is easier to define on itself. There still are different approaches, but for convex polyhedra they are all equivalent. In contrast to general polyhedra, there is general consensus about what are convex polyhedra.

Definition 2.14. A *convex polyhedron* P is the convex hull of a finite number of points in $\mathbb{R}^3$. The *extremal points* of P, i.e. the points that do not belong to a line segment in P, are the *vertices*. The 1-dimensional intersections with supporting hyperplanes are the *edges*, and the 2-dimensional intersections with supporting hyperplanes are the *faces*.

Equivalently, we can define a convex polyhedron as the intersection of a finite family of halfspaces [Grü03].

The vertices and edges of a convex polyhedron form a combinatorial graph that can be embedded in the plane by keeping the order of the edges around the vertices. This class of *polyhedral graphs* can be characterized by *Steinitz's theorem* [Ste16], as reformulated by Grünbaum [Grü03] who called it "the most important and deepest known result on 3-polytopes [i.e. convex polyhedra]".

Theorem 2.15 (Steinitz's theorem)**.** *The edges and vertices of a convex polyhedron form a 3-connected planar graph, and each 3-connected planar graph can be represented as the graph of a convex polyhedron.*

Furthermore, for each polyhedral graph G there exists a corresponding convex polyhedron P such that each automorphism of G is induced by a symmetry of P [Man71]. The plane embedding of a polyhedral graph is unique up to mirror symmetry [Whi32]. As a consequence of these results, we can identify polyhedra and 3-connected plane graphs. By doing that, we do not have to worry about geometric realisations.

6 Chamber systems

Every embedded graph G has an associated chamber system C_G [DH87]. This chamber system can be obtained by constructing a barycentric subdivision of G, i.e. subdividing each edge by one vertex in its center, adding one vertex in the center of each face, and adding edges from each center of a face to its vertices and centers of edges. Since we are working with combinatorially embedded graphs, the "center" can be chosen anywhere on the edge or in the face. This construction results in a plane triangulation. In C_G, each vertex v has a type $t(v) \in \{0, 1, 2\}$, indicating the dimension of its corresponding structure in G, i.e. type 0 for vertices, type 1 for edges and type 2 for faces. Each edge e has the type $t(e)$ of the opposite vertex in adjacent triangles, or equivalently the type that is different from the types of its endpoints. It is easily seen that this is well-defined, since adjacent vertices have distinct types. Conversely, the type of a vertex can be derived from its incident edges. See Figure 2.7 for an example.

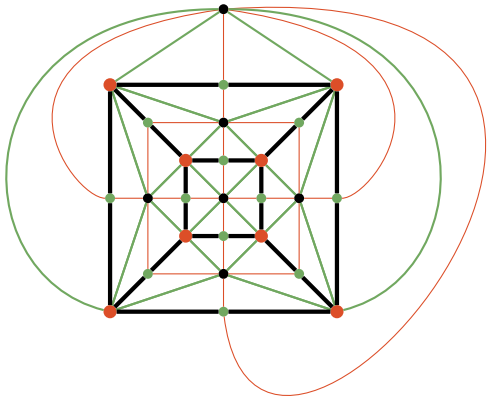


Figure 2.7. The chamber system of the cube. We will always use red for type 0, green for type 1, and black for type 2.

We call the triangular faces of C_G the *chambers*. Since each chamber of a 2-connected embedded graph is defined by a triplet of a vertex v of G, an edge e of G incident to v, and a face f of G incident to both v and e, the chambers of C_G are equivalent to the flags of G.

We call a pair of chambers sharing an edge of type 0 a *double chamber*. Each chamber of C_G is contained in exactly one double chamber.

The subgraph of a chamber system C_G that contains only the edges of type i and the vertices with a type different from i is called the *type-i subgraph*.

Theorem 2.16. *An embedded triangulation C with vertex set V and edge set E, together with a labelling function $t : V \cup E \to \{0,1,2\}$ is the chamber system of an embedded graph if and only if*

1. *for each edge $e = (v,w)$, $\{t(e), t(v), t(w)\} = \{0,1,2\}$;*
2. *for each vertex v with $t(v) = i$, the types of incident edges are alternating between j and k with $\{i,j,k\} = \{0,1,2\}$;*
3. *for each vertex v*

$$\begin{aligned} t(v) = 1 \quad &\Rightarrow \quad \deg(v) = 4 \\ t(v) \neq 1 \quad &\Rightarrow \quad \deg(v) > 4. \end{aligned}$$

Proof. It is easy to verify that the chamber system of an embedded graph satisfies these properties.

Given a triangulation C and a labelling function t that satisfy properties 1-3, we can remove each vertex v with $t(v) = 2$ together with its incident edges. The result is a graph with only edges of type 2. Since each vertex v with $t(v) = 1$ originally had degree 4, and the original incident edges were alternating between types 0 and 2, the current degree is 2. So we can remove all vertices of type 1, and combine their incident edges to one edge. The result is an embedded graph G with $C_G = C$. □

7 Connectivity

Since every polyhedron corresponds to a plane graph, we can construct its chamber system. But sometimes we want to go in the other direction, and derive the polyhedron from its chamber system. In that case, it is useful to know from the chamber system C_G whether the graph G is 3-connected (and thus a polyhedron). We will make use of *Menger's theorem* [Men27].

Theorem 2.17 (Menger's theorem)**.** *A graph is k-connected if and only if for every pair of vertices u and v there exist at least k internally disjoint paths between u and v.*

To decide whether a plane graph G is k-connected based on its chamber system C_G, we can look at the *type-1 cycles* in C_G. A type-1 cycle is a cycle in the type-1 subgraph of C_G. We call a type-1 cycle *empty* if there are no vertices on the inside or on the outside of the cycle in the type-1 subgraph. Note that in the graph C_G these cycles are not necessarily empty.

Lemma 2.18. *A plane graph G is*

1. *2-connected if and only if C_G contains no type-1 cycles of length 2;*
2. *3-connected if and only if G is 2-connected and C_G contains no non-empty type-1 cycles of length 4.*

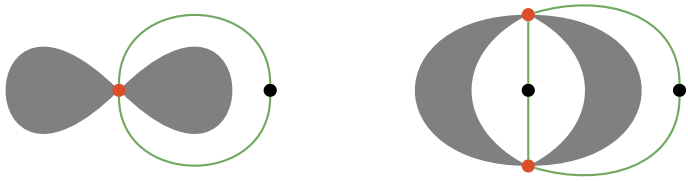
Proof.

1. Suppose C_G contains a type-1 cycle of length 2. This cycle contains one type-0 vertex v, incident to at least one type-2 edge inside the cycle and at least one type-2 edge outside the cycle (see Section 7), because C_G is a barycentric subdivision. It is clear that v has to be a cut-vertex of G.

   Conversely, if G has a cut-vertex v, there is a face of G for which v occurs at least two times in its border. In C_G this face corresponds to a type-2 vertex, incident to at least two type-1 edges to v. These edges form a type-1 cycle in C_G.

2. Suppose C_G contains a non-empty type-1 cycle of length 4, as can be seen in Section 7. This cycle contains two type-0 vertices v and w, with incident type-2 edges at both sides of the cycle. Removing v and w from G results in a disconnected graph.

   If G is 2-connected but not 3-connected, there are two vertices v and w that disconnect G when removed. So there are two non-empty subgraphs of G that are only connected by v and w, as in Section 7. This means that there is a non-empty type-1 cycle in C_G.



(a) not 2-connected (b) 2-connected but not 3-connected

Figure 2.8. Two graphs with type-1 cycles. The grey area contains the graph. Only the type-1 edges of the chamber system are shown.

□

References

[BM07] G. Brinkmann and B.D. McKay. “Fast generation of planar graphs”. In: *MATCH Communications in Mathematical and in Computer Chemistry* 58.2 (2007), pp. 323–357.

[CBG08] J.H. Conway, H. Burgiel and C. Goodman-Strauss. *The Symmetries of Things*. A K Peters, 2008.

[Die97] R. Diestel. *Graph Theory*. Graduate Texts in Mathematics 173. Springer-Verlag, 1997.

[DH87] A.W.M. Dress and D. Huson. “On tilings of the plane”. In: *Geometriae Dedicata* 24.3 (1987), pp. 295–310.

[GR01] C. Godsil and G. Royle. *Algebraic Graph Theory*. Graduate Texts in Mathematics 207. Springer, 2001.

[GT01] J.L. Gross and T.W. Tucker. *Topological Graph Theory*. Dover Publications, 2001.

[GYZ14] J.L. Gross, J. Yellen and P. Zhang, eds. *Handbook of Graph Theory*. 2nd ed. CRC Press, 2014.

[Grü94] B. Grünbaum. “Polyhedra with hollow faces”. In: *Proceedings of the NATO Advanced Study Institute on Polytopes: Abstract, Convex and Computational*. Ed. by T. Bisztriczky et al. Springer, 1994, pp. 43–70.

[Grü03] B. Grünbaum. *Convex Polytopes*. Ed. by V. Kaibel, V. Klee and G.M. Ziegler. 2nd ed. Graduate Texts in Mathematics 221. Springer, 2003.

[GS87] B. Grünbaum and G.C. Shephard. *Tilings and Patterns*. W. H. Freeman and Company, 1987.

[Man71] P. Mani. “Automorphismen von polyedrischen Graphen”. In: *Mathematische Annalen* 192 (1971), pp. 279–303.

[Men27] K. Menger. “Zur allgemeinen Kurventheorie”. In: *Fundamenta Mathematicae* 10.1 (1927), pp. 96–115.

[Ste16] E. Steinitz. “Polyeder und Raumeinteilungen”. In: *Encyclopädie der mathematischen Wissenschaften*. Ed. by W.F. Meyer and H. Mohrmann. Vol. 3.AB12. B.G. Teubner, 1916.

[Whi32] H. Whitney. “Congruent Graphs and the Connectivity of Graphs”. In: *American Journal of Mathematics* 54.1 (1932), pp. 150–168.

3

Local symmetry preserving operations

Although the operations described in Chapter 1 are often used in mathematics, there currently exists no systematic way to describe them. There exists an extensive naming scheme, and a lot of popular operations are well-studied, but without a formal definition it is unclear how many other operations there are and what they can or cannot do.

In this chapter, we introduce a formal definition of *local symmetry preserving operations*, generalizing the ideas of Goldberg [Gol37]. This definition includes all previously described symmetry preserving operations. The term *symmetry preserving* can be easily formalized: the symmetry group of the embedded graph to which the operation is applied must be a subgroup of the symmetry group of the embedded graph that is the result of the operation. The fact that the operations are defined on the level of a single chamber justifies the term *local*.

Note that not all operations – e.g. the chiral Goldberg operations – in Chapter 1 are symmetry preserving. We will extend our definition to include these in Chapter 5.

1 Operations and chamber decorations

Definition 3.1. Let T be a periodic tiling embedded in the plane, with points v_0 and v_2 such that v_0 is the center of a rotation ρ_{v_0} by 120 degrees in clockwise direction that is a symmetry of T, v_2 is the center of a rotation ρ_{v_2} by 60 degrees in clockwise direction that is a symmetry of T, and the line L_1 through v_0 and v_2 is a mirror axis of T.

We call (T, v_0, v_2) a *local symmetry preserving operation*, *lsp operation* for short.

The rotation $\rho_{v_1} = \rho_{v_2} \circ \rho_{v_0}$ is a rotation by 180 degrees, with a center which we will call v_1. The lines L_0 through v_1 and v_2, and L_2 through v_0 and v_1 are mirror axes of T. Note that v_0, v_1 and v_2 always are vertices of C_T, since only vertices, centers of edges and centers of faces can be the center of a rotation symmetry. A simple example of a periodic tiling T that satisfies these requirements, is the hexagonal tiling in Figure 3.1.

Definition 3.2. The subgraph D of the chamber system C_T bounded by the triangle between L_0, L_1 and L_2, together with the labelled vertices v_0, v_1 and v_2, is called the *chamber decoration* of the lsp operation (T, v_0, v_2).

We call the edges of D that coincide with L_i the *side* of type i, and the vertex v_i the *corner* of type i.

We can apply an lsp operation to an embedded graph or tiling G by subdividing each triangular chamber C of C_G with the chambers of D, identifying vertex v_i with the corner of C with type i. The resulting graph satisfies the properties of Theorem 2.16, so it is the chamber system $C_{G'}$ of an embedded graph G'. In Figure 3.2, the lsp operation ambo is applied to the cube.

The choice of 30 and 60 degrees in Definition 3.1 is a reference to Goldberg's paper. One could also interchange the values, require 45 degrees for both or even choose other values and go to hyperbolic or spherical instead of plane tilings and still get the same set of operations

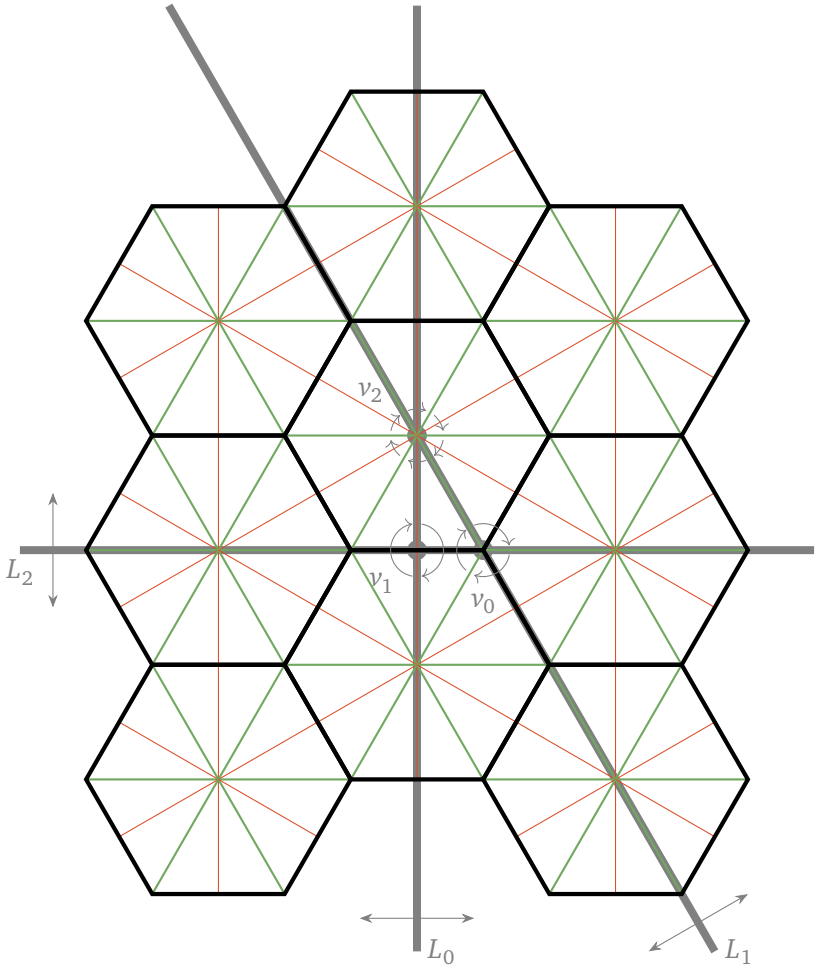


Figure 3.1. The barycentric subdivision of a hexagonal tiling, with v_0, v_1 and v_2 as in Definition 3.1.

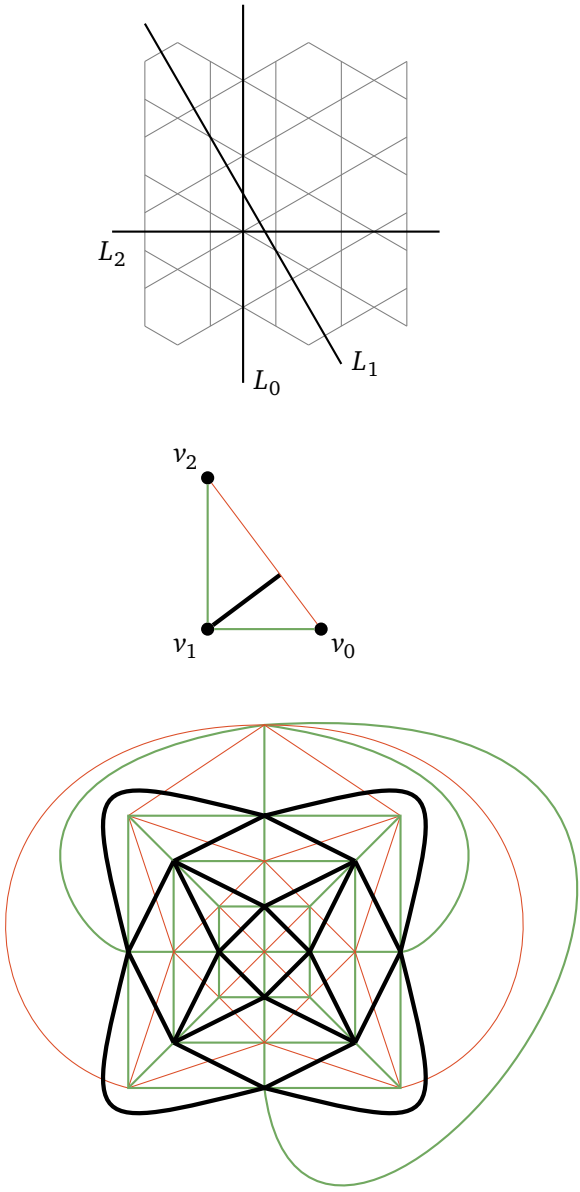


Figure 3.2. The lsp operation ambo, its chamber decoration, and application to the cube (see Figure 2.7). The resulting graph is the one in black, which is a cuboctahedron.

(see Figure 3.3). Each lsp operation that can be derived with 30 and 60 degrees angles, can be derived from these other angles too and vice versa.

An alternative way to understand that different choices of angles give the same set of operations is as follows: Let T_0 be a tiling of the sphere or the Euclidean or hyperbolic plane where the symmetry group acts transitively on the chambers. Then any lsp operation can be applied to T_0 to give another tiling T_1, from which the operation can be recovered by requiring the angles of the chambers of T_0 in the definition. We can for example apply the lsp operation ambo (see Figure 3.2) to a square tiling T_0, which results in the tiling T_1 in Figure 3.4. We can now require two rotations of 90 degrees in the definition, and choose v_0 and v_2 so that we get the same operation.

Each lsp operation corresponds to exactly one chamber decoration. Since chamber decorations are easier to draw and understand, we will usually identify an lsp operation with its chamber decoration.

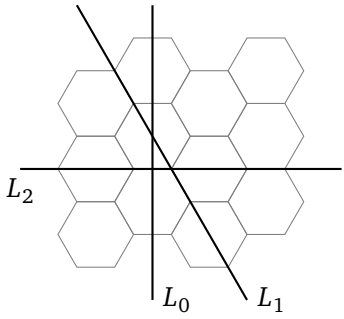
Theorem 3.3. *A 2-connected plane graph D with vertex set V and edge set E, together with a labelling function $t : V \cup E \to \{0,1,2\}$, and an outer face which contains vertices v_0, v_1, v_2, is a chamber decoration if and only if*

1. *all inner faces are triangles;*
2. *for each edge $e = (v,w)$, $\{t(e), t(v), t(w)\} = \{0,1,2\}$;*
3. *for each vertex v with $t(v) = i$, the types of incident edges are j and k with $\{i,j,k\} = \{0,1,2\}$. Two consecutive edges with an inner face in between them cannot have the same type;*
4. *for each inner vertex v*

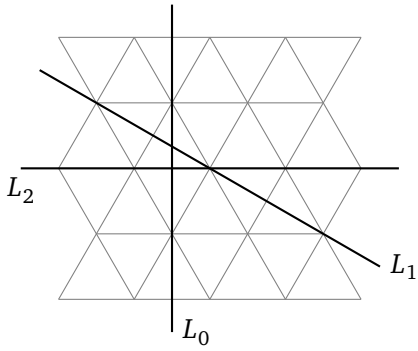
$$\begin{aligned} t(v) = 1 \quad &\Rightarrow \quad \deg(v) = 4 \\ t(v) \neq 1 \quad &\Rightarrow \quad \deg(v) > 4, \end{aligned}$$

for each vertex v in the outer face and different from v_0, v_1, v_2

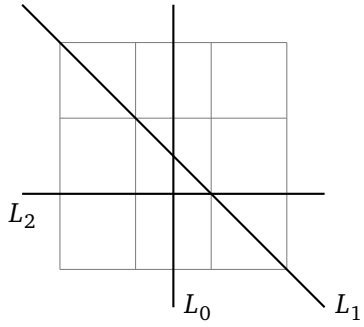
$$\begin{aligned} t(v) = 1 \quad &\Rightarrow \quad \deg(v) = 3 \\ t(v) \neq 1 \quad &\Rightarrow \quad \deg(v) > 3 \end{aligned}$$



(a) 30° and 60°



(b) 60° and 30°



(c) 45° and 45°

Figure 3.3. Identity defined with different angles.

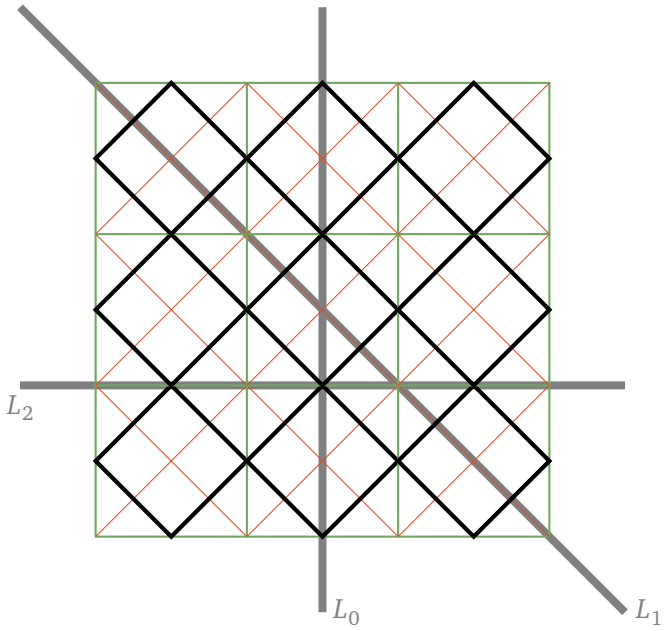


Figure 3.4. Ambo defined with two 90° angles.

and

$$\begin{aligned} & t(v_0), t(v_2) \neq 1 \\ t(v_1) = 1 \quad & \Rightarrow \quad \deg(v_1) = 2 \\ t(v_1) \neq 1 \quad & \Rightarrow \quad \deg(v_1) > 2. \end{aligned}$$

Proof. It is easily verified that each chamber decoration satisfies the given properties. We still have to prove that each graph D with these properties is the chamber decoration of an lsp operation.

We take the hexagonal tiling H and use D to decorate each chamber of the chamber system C_H. The result will be a chamber system C_T of a tiling T by Theorem 2.16.

We will first prove that the type-2 subgraph of D is connected, by induction on the number of triangles. There is always at least one triangle in D that shares one or two edges with the outer face. We remove these edges, and call the result D'. It is clear that D' still satisfies

properties 1–3, and by induction its type-2 subgraph is connected. If one of the removed edges has type 2, it is connected to D' by a vertex of type 0 or 1 with degree at least 3, and therefore it is connected to the type-2 subgraph of D'.

Given vertices u and v in the type-2 subgraph of C_T, there exists a sequence of chambers $C_0, \ldots, C_n$ of H such that two consecutive chambers C_i and C_{i+1} share one side, and u is contained in C_0 and v in C_n. Since there are at least two vertices on each side of D, and they are not both of type 2, at least one of them is in the type-2 subgraph of C_T. Thus, there is a type-2 path between u and v that passes through all chambers in the sequence $C_0, \ldots, C_n$, and the type-2 subgraph of C_T is connected. It follows immediately that T is connected too.

We can choose the vertices of one chamber of C_H in T as v_0, v_1 and v_2. Now, (T, v_0, v_2) is an lsp operation according to Definition 3.1, and it is clear that the corresponding chamber decoration is equal to D. □

2 Connectivity

Since we are primarily interested in operations on polyhedra, which are 3-connected plane graphs, we want to identify the lsp operations that result in a polyhedron when applied to a polyhedron.

We call an lsp operation *k-connected* for $k \in \{1, 2, 3\}$ if it is derived from a k-connected tiling T.

Theorem 3.4. *If G is a k-connected plane graph with $k \in \{1, 2, 3\}$, and O is a k-connected lsp operation, then $O(G)$ is a k-connected plane graph.*

Proof. It is clear that $O(G)$ is a plane graph. For $k = 1$, we know that T and G are connected, and it follows easily that $O(G)$ is connected.

For $k = 2$, we will prove that there is no cut-vertex in $O(G)$. A type-1 cycle of length 2 in $C_{O(G)}$ is either completely contained in one chamber of C_G[1], or it is split between two chambers of C_G (see Figure 3.5).

[1]With a chamber of C_G in $C_{O(G)}$, we mean the area that was a chamber of C_G before it was subdivided by O

Both cases cannot appear, as for any chamber (resp. any pair of adjacent chambers) there is an isomorphism between this chamber (resp. these two chambers) and the corresponding area in T, and according to Lemma 2.18 T has no type-1 cycles of length 2.

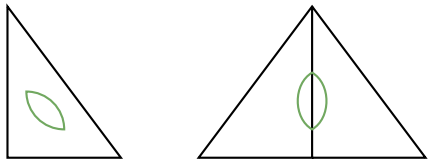


Figure 3.5. The different situations where type-1 cycles of length 2 can occur.

This implies that $C_{O(G)}$ contains no type-1 cycles of length 2, and thus, invoking once again Lemma 2.18, $O(G)$ contains no cut-vertices.

For $k = 3$, we will prove that $O(G)$ is 3-connected. Consider a type-1 cycle C with edges $e_1, \ldots, e_n$ in $C_{O(G)}$. Let $C_1, \ldots, C_n$ be a sequence of double chambers of C_G such that e_i is contained in C_i for $1 \leq i \leq n$. If $C_i = C_{i+1}$, we can remove C_i from the sequence. This results in the reduced sequence $C_1, \ldots, C_m$.

If C has length 2, then $m \leq 2$. Thus, the cycle is contained in one or two neighbouring chambers of C_G, and it should be present in the tiling T too, which is impossible.

If C has length 4, then $m \leq 4$. Thus, the cycle is contained in at most 4 chambers of C_G, and each chamber has at least one vertex or edge in common with the previous and next one, but not the same for both of them. Checking the different configurations of at most four chambers in a chamber system that have these properties, one can prove that there is a type-1 cycle C' of length at most 4 in C_G with an edge in each of $C_1, \ldots, C_m$. Since G is 3-connected, C' cannot have length 2, and if C' has length 4 it has to be empty. Thus, the situation is as in Figure 3.6, and cycle C lies completely in the chambers adjacent to C'. Since there is an isomorphism between this area in C_G and the corresponding area in C_T, and T is 3-connected, C cannot be a non-empty type-1 cycle. Therefore, $O(G)$ is 3-connected because of Lemma 2.18.

□

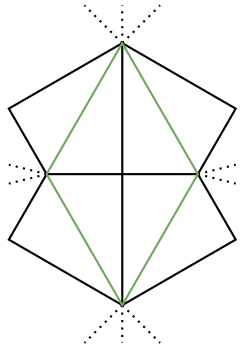


Figure 3.6. The type-1 cycle C' with adjacent chambers.

Remark 3.5. This theorem only holds for plane graphs, since the proof relies on the Jordan curve theorem. A counterexample to an equivalent theorem for embedded graphs of higher genus is the dual of a 3-connected graph on the torus, which can have a 2-cut [BBZ18].

The connectivity of an lsp operation can be derived from its chamber decoration.

Lemma 3.6. *A chamber decoration D is 2-connected if and only if it has*

1. *no type-1 cycles of length 2;*
2. *no internal type-1 edges between two vertices on a single side.*

Proof. A chamber decoration D that satisfies these properties defines an lsp operation according to Theorem 3.3. We will prove that the corresponding tiling T is 2-connected. If T is not 2-connected, there is a type-1 cycle of length 2 in C_T. If this cycle is completely contained in the triangle v_0, v_1, v_2, there is a cycle of length 2 in D too, which is impossible. The only other possibility is that the cycle of length 2 is cut in half by L_i, but then there is an internal type-1 edge between 2 vertices on L_i, which is a side of D.

A 2-connected lsp operation with corresponding tiling T defines a decoration D according to Theorem 3.3. If there is a type-1 cycle of length 2 in D, this cycle occurs in C_T too, and T is not 2-connected. If

there is an internal type-1 edge between 2 vertices on the same side, this will result in a cycle of length 2 in T because this side lies on a mirror axis of T. Therefore, the properties are satisfied. □

Lemma 3.7. *A chamber decoration is 3-connected if and only if it is 2-connected and has*

1. *no type-1 edge between sides 0 and 2;*
2. *no non-empty type-1 cycles of length 4.*

Proof. A 2-connected chamber decoration that satisfies these properties defines a 2-connected lsp operation according to Lemma 3.6. If T is not 3-connected, there is a non-empty type-1 cycle of length 4. If this cycle is completely contained in the triangle v_0, v_1, v_2, as in Section 2, there is a type-1 cycle of length 4 in D. If the cycle is cut in half by L_i, as in Section 2, there is an internal type-1 path of length 2 between 2 vertices on L_i, which is a side of D. If the cycle is cut in four, as in Section 2, there is a type-1 edge between sides 0 and 2.

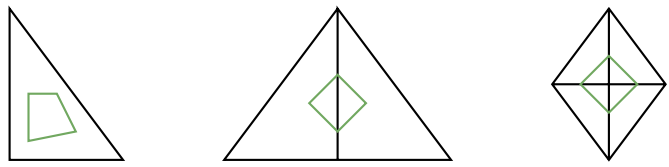


Figure 3.7. The different situations where non-empty type-1 cycles of length 4 can occur.

A 3-connected lsp operation with corresponding tiling T defines a 2-connected decoration D according to Lemma 3.6. If there is a type-1 cycle of length 4 in D, this cycle occurs in C_T too, and T would not be 3-connected. If there is an internal type-1 path of length 2 between 2 vertices on the same side, or a type-1 edge between sides 0 and 2, this will result in a cycle of length 4 in T. □

3 Composition

In this section, we investigate the algebraic properties of lsp operations. We start with the definition of a composition operation for lsp operations.

Definition 3.8. Given two lsp operations o_1 and o_2 derived from tilings T_1 and T_2, we can apply the operation o_2 to T_1. This will result in a connected tiling T', and since symmetries are preserved the points v_0 and v_2 will have the same rotation symmetry, and the line L_1 will have the same mirror symmetry as in T_1. We call the lsp operation $o_2o_1 = (T', v_0, v_2)$ the *composition* of o_1 and o_2.

The chamber decoration of the composition o_2o_1 can be obtained by "applying" o_2 to the chamber decoration of o_1 (but only to the internal chambers, keeping the outer face untouched).

Theorem 3.9. *The set of lsp operations together with the composition operation forms a monoid, i.e.*

$$a(bc) = (ab)c$$
$$ia = ai = a$$

for all lsp operations a, b, c and identity i.

Proof. The chamber decoration $D_{a(bc)}$ is obtained by decorating each chamber of D_c with D_b, and subsequently decorating each resulting chamber with D_a. This is the same as decorating each chamber of D_b with D_a to get D_{ab}, and subdividing each chamber of D_c with D_{ab}. Therefore, $D_{a(bc)} = D_{(ab)c}$.

The identity element i for composition is the identity operation. □

Theorem 3.10. *The lsp operations define a* monoid action *on the embedded graphs, i.e.*

$$(ab)G = a(bG)$$
$$iG = G$$

for all lsp operations a and b, embedded graph G and identity operation i.

Proof. The graph $(ab)G$ is constructed by decorating each chamber of G with the chamber decoration D_{ab}, and D_{ab} is equal to D_b with each chamber decorated by D_a. The graph $a(bG)$ is constructed by decorating each chamber of G with D_b, and subsequently decorating each chamber with D_a. It is clear that the result is the same.

It is trivial that $iG = G$ if i is identity. □

The only invertible lsp operations are identity and dual, and both of them are their own inverse. This will become clear in Lemma 3.14.

Definition 3.11. A *simple* lsp operation o is an lsp operation that cannot be written as the composition of two lsp operations a and b different from identity and dual.

Each lsp operation o can be composed with dual to get four related lsp operations o, do, od and dod. An lsp operation is simple if and only if the related lsp operations are simple. For some lsp operations, $o = od$, in which case there are only two related lsp operations.

4 Inflation

If we want to measure the impact of an lsp operation on the "size" of an embedded graph, we need an invariant property of the lsp operation that is independent of the graph to which it is applied. We write v_G, e_G and f_G for resp. the number of vertices, edges and faces of an embedded graph G.

If the chamber decoration of an lsp operation o has r chambers, the number of chambers of $o(G)$ is r times the number of chambers of G. Since each edge of an embedded graph G is surrounded by four chambers in C_G and each chamber determines exactly one edge, the number of chambers of G and $o(G)$ are resp. $4e_G$ and $4e_{o(G)}$. Therefore, the number of edges increases with a factor r, i.e. $e_{o(G)} = re_G$. In [PWB17], this factor is called the *edge-multiplier*.

Definition 3.12. The *inflation factor* $r(o)$ of an lsp operation o is the number of edges of $o(G)$ divided by the number of edges of G, for any plane graph G.

It follows immediately that the inflation factor of an lsp operation o is equal to the number of chambers of the chamber decoration of o.

Theorem 3.13. *The inflation factor $r(o)$ of an lsp operation o is a representation of the monoid of lsp operations.*

$$r(i) = 1$$
$$r(ab) = r(a)r(b)$$

Proof. The identity chamber decoration has obviously only one chamber, so $r(i) = 1$.

Since the chamber decoration D_{ab} is obtained by replacing each of the $r(b)$ chambers of D_b with the $r(a)$ chambers of D_a, it will have $r(a)r(b)$ chambers. □

Lemma 3.14. *The only lsp operations with inflation factor 1 are identity and dual. Consequently, they are the only invertible lsp operations.*

Proof. The chamber decoration of an lsp operation with inflation factor 1 has exactly one chamber. There are only two possibilities, and these are the chamber decorations of identity and dual.

For an invertible lsp operation o, we have $o^{-1}o = i$, and therefore $r(o^{-1})r(o) = 1$. Since the inflation factor of an lsp operation is integral, this implies $r(o) = r(o^{-1}) = 1$. □

Although the number of vertices and faces $v_{o(G)}$ and $f_{o(G)}$ are not multiples of v_G resp. f_G, we can prove that they are linear combinations of v_G, e_G and f_G. So we can extend the notion of inflation factor to an *inflation matrix*, as observed by [Rec17].

Suppose D is the chamber decoration with r chambers of an lsp operation o. If a_i is the number of inner vertices of type i in D, and b_i is the number of vertices of type i in the outer face of D excluding the three corner vertices, we define

$$d_i = a_i + \frac{b_i}{2}.$$

If the type-j corner of D has type i, we define $c_{i,j} = 1$, and else $c_{i,j} = 0$. Note that for any $j \in 0, 1, 2$, exactly one of $c_{0,j}$, $c_{1,j}$ and $c_{2,j}$

is equal to one, and since the type of neither v_0 or v_2 is 1 according to Theorem 3.3, $c_{1,0}$ and $c_{1,2}$ are zero. In summary,

$$\begin{aligned} c_{0,0} + c_{2,0} &= 1 \\ c_{0,1} + c_{1,1} + c_{2,1} &= 1 \\ c_{0,2} + c_{2,2} &= 1. \end{aligned}$$

Lemma 3.15. *The matrix*

$$M_o = \begin{pmatrix} r & 0 & 0 & 0 \\ d_0 & c_{0,0} & c_{0,1} & c_{0,2} \\ d_1 & c_{1,0} & c_{1,1} & c_{1,2} \\ d_2 & c_{2,0} & c_{2,2} & c_{2,2} \end{pmatrix},$$

is a monoid representation, i.e. $M_{o'o} = M_{o'}M_o$, *and*

$$\begin{pmatrix} c_{o(G)} \\ v_{o(G)} \\ e_{o(G)} \\ f_{o(G)} \end{pmatrix} = M_o \begin{pmatrix} c_G \\ v_G \\ e_G \\ f_G \end{pmatrix}$$

for every embedded graph G with c_G chambers.

Proof. Given two lsp operations o and o', with corresponding chamber decorations D and D', and numbers $r, d_i, c_{i,j}$ resp. $r', d'_i, c'_{i,j}$, we can compute the numbers $r'', d''_i, c''_{i,j}$ of the lsp operation $o'' = o'o$ with chamber decoration D'' (see Figure 3.8 for an example).

The number of chambers r'' in D'' is equal to $r'r$ because of Theorem 3.13. Each of the r chambers of D is replaced with D', which has a'_i inner vertices and b'_i vertices on the border of type i. Hence, the number a''_i of inner vertices of D'' of type i is at least ra'_i. The remaining vertices of type i are vertices on the borders and corners of D'. Those on the borders always occur on the border of two neighbouring chambers, or one chamber if they occur on the border of D''. So there are $\frac{rb'_i}{2}$ of them if we count these on the border of D'' only half. If the type-j corner of D' has type i — and therefore $c'_{i,j} = 1$ — there are a_j extra inner vertices and b_j vertices on the border of D'' of type i. Finally, the vertex on the type-j corner of D'' has type i if

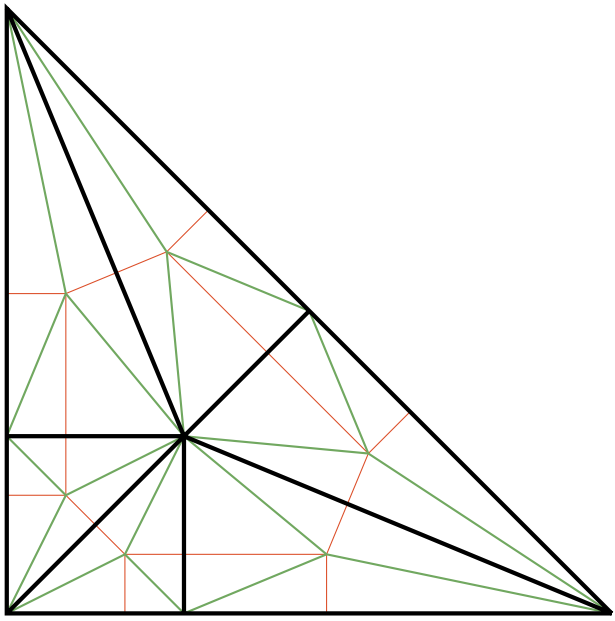


Figure 3.8. The operation $m^2 = mm$, i.e. meta applied to itself. For meta, we have $r = 6$, $d_0 = 0$, $d_1 = 1.5$, $d_2 = 1$ and $c_{0,1} = c_{0,2} = c_{0,3} = 1$. For the resulting lsp operation m^2, we have $r' = 36$, $d'_0 = 2.5$, $d'_1 = 9$, $d'_2 = 6$ and $c_{0,1} = c_{0,2} = c_{0,3} = 1$.

there exists a $k \in \{0, 1, 2\}$ such that the type-j corner of D has type k and the type-k corner of D' has type i. In that case, there is exactly one such k. In summary,

$$\begin{aligned} r'' &= r'r \\ d''_i &= ra'_i + \frac{rb'_i}{2} + \sum_j c'_{i,j}a_j + \frac{c'_{i,j}b_j}{2} \\ &= rd'_i + \sum_j c'_{i,j}d_j \\ c''_{i,j} &= \sum_k c'_{i,k}c_{k,j}, \end{aligned}$$

which is equivalent to $M_{o'o} = M_{o'}M_o$. □

Since the number of chambers c_G can be computed from v_G, e_G and f_G, the matrix M_o contains redundant information.

Definition 3.16. The matrix

$$R(o) = \begin{pmatrix} c_{0,0} & 4d_0 + c_{0,1} & c_{0,2} \\ 0 & 4d_0 + 4d_2 + 2 - c_{1,1} & 0 \\ 1 - c_{0,0} & 4d_2 + 1 - c_{0,1} - c_{1,1} & 1 - c_{0,2} \end{pmatrix}$$

is the *inflation matrix* of an lsp operation o.

Theorem 3.17. *The inflation matrix $R(o)$ is a monoid representation, i.e. $R(o'o) = R(o')R(o)$, and*

$$\begin{pmatrix} v_{o(G)} \\ e_{o(G)} \\ f_{o(G)} \end{pmatrix} = R(o) \begin{pmatrix} v_G \\ e_G \\ f_G \end{pmatrix}.$$

for every embedded graph G.

Proof. Suppose v_D, e_D and f_D are the numbers of vertices, edges and inner faces of the chamber decoration D, and b_D is the size of the outer face. Then

$$\begin{aligned} f_D &= r, \\ v_D &= d_0 + d_1 + d_2 + \frac{b_D + 3}{2}, \end{aligned}$$

and according to Euler's formula (Definition 2.8) and since D is a plane graph,

$$e_D = v_D + f_D - 1 = d_0 + d_1 + d_2 + \frac{b_D + 1}{2} + r.$$

But since every inner face of D is triangular, we can alternatively count the number of edges as

$$e_D = \frac{3f_D + b_D}{2} = \frac{3r + b_D}{2},$$

and therefore

$$d_0 + d_1 + d_2 + \frac{b_D + 1}{2} + r = \frac{3r + b_D}{2},$$

which gives $r = 2(d_0 + d_1 + d_2) + 1$.

Since the number of chambers of an embedded graph is four times the number of edges, according to Lemma 3.15

$$\begin{aligned} e_{o(G)} &= d_1 c_G + c_{1,0} v_G + c_{1,1} e_G + c_{1,2} f_G \\ &= 4d_1 e_G + c_{1,1} e_G \\ &= (4d_1 + c_{1,1}) e_G. \end{aligned}$$

But we already know that $e_{o(G)} = r e_G$, so

$$4d_1 + c_{1,1} = r = 2(d_0 + d_1 + d_2) + 1,$$

which results in $d_1 = d_0 + d_2 + \frac{1-c_{1,1}}{2}$.

So we can rewrite the matrix M_o as

$$M_o = \begin{pmatrix} r & 0 & 0 & 0 \\ d_0 & c_{0,0} & c_{0,1} & c_{0,2} \\ d_0 + d_2 + \frac{1-c_{1,1}}{2} & 0 & c_{1,1} & 0 \\ d_2 & 1-c_{0,0} & 1-c_{0,1}-c_{1,1} & 1-c_{0,2} \end{pmatrix}.$$

If we use this matrix to compute the numbers of vertices, edges and faces of $o(G)$, we get

$$\begin{aligned} v_{o(G)} &= d_0 c_G + c_{0,0} v_G + c_{0,1} e_G + c_{0,2} f_G \\ &= 4d_0 e_G + c_{0,0} v_G + c_{0,1} e_G + c_{0,2} f_G \\ &= c_{0,0} v_G + (4d_0 + c_{0,1}) e_G + c_{0,2} f_G \\ e_{o(G)} &= (d_0 + d_2 + \frac{1-c_{1,1}}{2}) c_G + c_{1,1} e_G \\ &= 4(d_0 + d_2 + \frac{1-c_{1,1}}{2}) e_G + c_{1,1} e_G \\ &= (4d_0 + 4d_2 + 2 - c_{1,1}) e_G \\ f_{o(G)} &= d_2 c_G + (1-c_{0,0}) v_G + (1-c_{0,1}-c_{1,1}) e_G + (1-c_{0,2}) f_G \\ &= 4d_2 e_G + (1-c_{0,0}) v_G + (1-c_{0,1}-c_{1,1}) e_G + (1-c_{0,2}) f_G \\ &= (1-c_{0,0}) v_G + (4d_2 + 1 - c_{0,1} - c_{1,1}) e_G + (1-c_{0,2}) f_G. \end{aligned}$$

So $v_{o(G)}$, $e_{o(G)}$ and $f_{o(G)}$ are a linear combination of v_G, e_G and f_G. If we write this as a matrix multiplication, we get

$$\begin{pmatrix} v_{o(G)} \\ e_{o(G)} \\ f_{o(G)} \end{pmatrix} = R(o) \begin{pmatrix} v_G \\ e_G \\ f_G \end{pmatrix}.$$

We can use the inflation matrix to prove the following lemma.

Lemma 3.18. *An lsp operation preserves the Euler characteristic, i.e.*

$$\chi(o(G)) = \chi(G)$$

for any lsp operation o.

Proof.

$$\begin{aligned}\chi(o(G)) &= v_{o(G)} - e_{o(G)} + f_{o(G)} \\ &= c_{0,0}v_G + (4d_0 + c_{0,1})e_G + c_{0,2}f_G - (4d_0 + 4d_2 + 2 - c_{1,1})e_G \\ &\quad + (1 - c_{0,0})v_G + (4d_2 + 1 - c_{0,1} - c_{1,1})e_G + (1 - c_{0,2})f_G \\ &= v_G - e_G + f_G \\ &= \chi(G)\end{aligned}$$

□

5 Achiral Conway operations

Now that we have a sound definition of lsp operations, we can revisit the Conway operations. Of course, we want that every Conway operation is covered by our definition. For the basic Conway operations, this is the case. But some of the extended operations are chiral, which means that they do not preserve orientation-reversing symmetries. We will ignore these for now, and only focus on the achiral ones. In Table 3.1 the chamber decorations of the achiral Conway operations are given.

For each Conway operation in Table 3.1, the four related lsp operations are given. Some of these are Conway operations too, but others have no name yet. A natural question is whether there are other simple lsp operations with small inflation rates that are not present in this list. To answer that question, we need a way to construct all lsp operations with a given inflation factor, which we will do in Chapter 4.

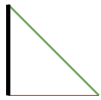
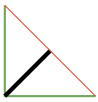
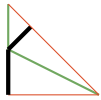
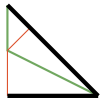
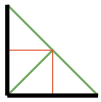
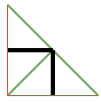
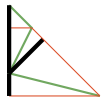
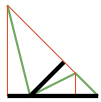
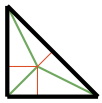
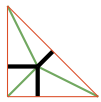
inflation factor	x	xd	dx	dxd
1	 identity	 dual		
2	 ambo		 join	
3	 truncate	 zip	 needle	 kis
4	 ortho		 expand	
	 chamfer			 subdivide
5	 loft			
6	 meta		 bevel	

Table 3.1. The achiral Conway operations. Related lsp operations obtained by composition with dual are on the same line.

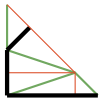
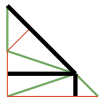
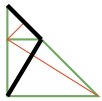
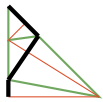
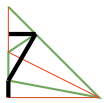
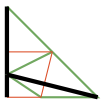
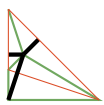
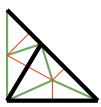
inflation factor	x	xd	dx	dxd
	 quinto			
	 join-lace			
7	 lace			
	 stake			
	 medial			
8	 join-kis-kis			
10	 cross			

Table 3.1. (continued)

Since we defined equality and composition for lsp operations, we can now prove the relations between the Conway operations. Since the inflation factor is a monoid representation, lsp operations with prime inflation factor are simple. For composite inflation rates, e.g. $r(o) = 4$, we only have to consider lsp operations with inflation rates that are a divisor of $r(o)$. The only composite Conway operations are ortho, expand, meta and bevel.

$$o = ja \qquad e = aa$$
$$m = na \qquad b = ta$$

6 Other operations

Some operations closely related to Conway operations are no lsp operations. The operation *one-dimensional subdivision* [OPW10] on maps is local and symmetry preserving, but introduces vertices of degree 2. Therefore, it is not an operation on polyhedra, and less interesting to us, although it is possible to expand Definition 3.1. But if we want to keep the monoid of lsp operations closed under composition, we have to allow the dual operation too, which introduces double edges. Since one-dimensional subdivision does not change the topological representation of a graph, we chose not to include it. One downside is that some simple lsp operations can be decomposed by using one-dimensional subdivision.

Another common operation is *alternation* [Cox73], which removes alternate vertices, such that each edge has one endpoint removed. Since this operation can only be applied to bipartite embedded graphs (e.g. polyhedra with only even-sized faces), this is not an lsp operation.

The *Petrie dual* is a symmetry preserving operation that can be applied to any embedded graph, but generally changes the surface on which the graph can be embedded, so it does not preserve the Euler characteristic.

References

[BBZ18] D. Bokal, G. Brinkmann and C.T. Zamfirescu. *The Connectivity of the Dual*. 2018. arXiv: 1812.08510.

[Cox73] H.S.M. Coxeter. *Regular Polytopes*. 3rd ed. Dover Publications, 1973.

[Gol37] M. Goldberg. "A Class of Multi-Symmetric Polyhedra". In: *Tohoku Mathematical Journal*. First Series 43 (1937), pp. 104–108.

[OPW10] A. Orbanić, D. Pellicer and A.I. Weiss. "Map operations and k-orbit maps". In: *Journal of Combinatorial Theory, Series A* 117.4 (2010), pp. 411–429.

[PWB17] T. Pisanski, G. Williams and L.W. Berman. "Operations on Oriented Maps". In: *Symmetry* 9 (2017), p. 274.

[Rec17] B.R.S. Recht. *Notes on operations on polyhedra*. 2017. URL: https://antitile.readthedocs.io/en/latest/conway.html.

4

Generation of local symmetry preserving operations

In the previous chapter we established a general framework in which the class of all lsp operations can be studied, without having to consider individual operations separately. It was shown that many of the most frequently used operations on polyhedra fit into this framework.

But of course we sometimes do want to examine the operations individually, e.g. to check conjectures on as many examples as possible before we try to prove them, or to find operations with certain properties. We can do this for a few operations by hand, but a computer can do this a lot faster, and in a systematic way such that no operations are missed.

In this chapter we will introduce an algorithm that generates all lsp operations. We will start with the generation of *predecorations*, using the *canonical construction path* method as described in [McK98]. Next, we can complete each of these predecorations in every possible way to a chamber decoration. According to the *homomorphism principle* [GLM97], this generates every chamber decoration, and therefore each lsp operation.

1 Predecorations

The generation of all chamber decorations will be split into two phases. In the first phase, we will construct the type-1 subgraph, consisting of all edges of type 1.

Let n_A be the number of vertices in the type-1 subgraph of degree 1 with a neighbouring vertex of degree 2, n_B the number of remaining vertices of degree 1, and n_C the number of quadrangles with three vertices of degree 2.

(a) n_A (b) n_B (c) n_C

Figure 4.1. The subgraphs counted as n_A, n_B and n_C.

Lemma 4.1. *Let D be a chamber decoration. The type-1 subgraph D_1 of D has the following properties:*

1. *all inner faces are quadrangles;*
2. *each inner vertex has degree at least 3;*
3. *$n_A \leq 2$ and $n_A + n_B + n_C \leq 3$.*

Proof. It follows immediately from the properties of a chamber decoration (Theorem 3.3) that the inner faces of D_1 are quadrangles and the inner vertices have degree at least 3.

Each area bounded by a quadrangle in D_1 contains one vertex of type 1 in D. The only other difference between D and D_1 is in the outer face of D_1, where type-1 vertices of degree 3 in D (a 3-completion), and at most one of degree 2 in D (a 2-completion), can be present in D. If there is a type-1 vertex of degree 2, then that vertex is v_1. An example can be seen in Figure 4.2.

The subgraph in Section 1 can only occur if the rightmost vertex v of degree two is v_0, v_1 or v_2, or if v_1 is a type-1 vertex of degree 2 connected to this vertex. Each of the three vertices of degree 2 in this subgraph of D_1 corresponds to v_0, v_1, v_2 or a vertex of degree at least

4 in D. The inner edges of the quadrangle in D contribute exactly one to the degree of these vertices. This implies that either there is a 2-completion here (in which case v_1 is connected to v), or there are two 3-completions which do not involve v (in which case v is v_0, v_1 or v_2).

The subgraph in Section 1 can only occur if the rightmost vertex v is v_0, v_1 or v_2. This vertex of degree 1 in D_1 corresponds to a vertex of degree at most 3 in D, which is only possible in v_0, v_1 or v_2.

The subgraph in Section 1 can only occur if the rightmost vertex v is v_0 or v_2. There are two neighbouring cut-vertices of D_1 in this subgraph, which do not correspond to cut-vertices in D. This is only possible if both of these vertices are the middle vertex of a 3-completion. This increases the degree of v in D to 2, which is only possible in v_0 or v_2. The degree of v can be 3 if there is a 2-completion too, but then v_1 is contained in this 2-completion and v still has to be v_0 or v_2.

We find that $n_A \leq |\{v_0, v_2\}| = 2$ and $n_A + n_B + n_C \leq |\{v_0, v_1, v_2\}| = 3$. □

Definition 4.2. A *predecoration* is a connected plane graph with an outer face that satisfies the properties of Lemma 4.1.

Given a predecoration P, we can try to add edges, vertices and labels to get a chamber decoration with P as its type-1 subgraph. We will have to add one type-1 vertex in each inner face of P, as in Figure 4.2. Then we can add type-1 vertices in the outer face, and connect them to three consecutive vertices of P. Finally, we can add a type-1 vertex in the outer face and connect it to two consecutive vertices of P. This vertex has to be v_1.

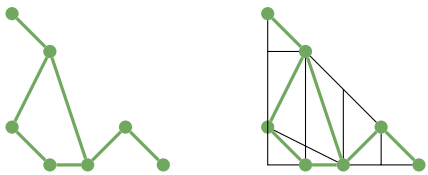


Figure 4.2. A predecoration with a possible completion.

By definition, the type-1 subgraph of a chamber decoration D is a predecoration. Unfortunately, not each predecoration corresponds to a type-1 subgraph of some chamber decoration. This is e.g. the case if there are too many cut-vertices, as in Figure 4.3.

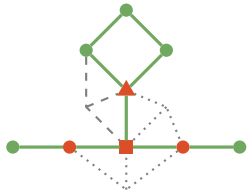


Figure 4.3. A predecoration with $n_A = 2$ and $n_C = 1$ and 4 cut-vertices that cannot be completed. In order to remove the triangular cut-vertex, the dashed edges have to be added. The square cut-vertex requires the dotted edges. But now the circular cut-vertex on the right cannot be removed by this completion.

2 Construction of predecorations

All predecorations can be constructed from the base decorations K_2 and C_4 (see Figure 4.4) using the 10 extension operations shown in Figure 4.5. We will prove this by showing that each predecoration, with the exception of K_2 and C_4, can be reduced by the inverse of one of the extension operations, which we call *reductions*. We will then use the *canonical construction path* method [McK98] to generate all predecorations without isomorphic copies.

Figure 4.4. The base predecorations.

Given a predecoration P, we will choose a *canonical parent* of P. This is a predecoration obtained by applying one of the reductions to P. We will always use the reduction with the smallest number among all possible reductions. It is possible that there is more than one way to apply this reduction to P, and if P has non-trivial symmetry, some of these can result in isomorphic parents.

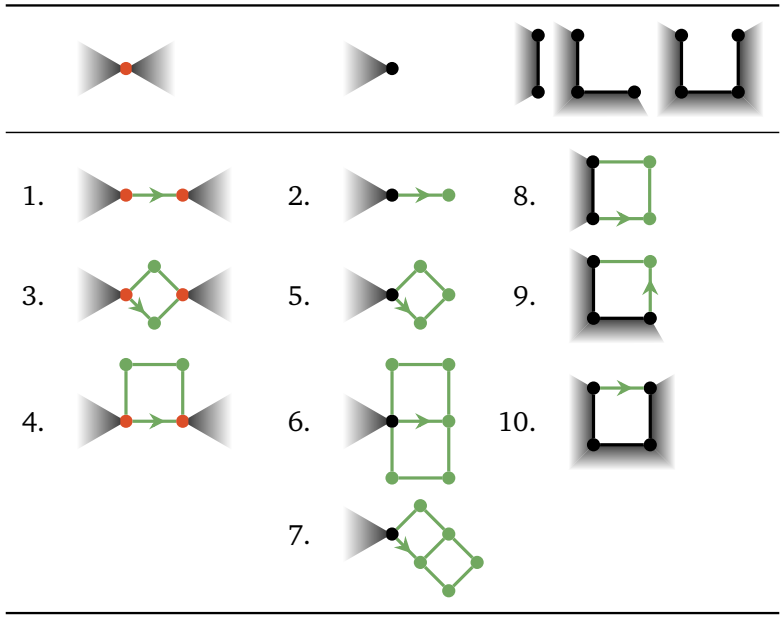


Figure 4.5. The extensions. In the first row, the subgraphs before the extension is applied are given. New edges and vertices are green, and vertices that are broken apart in two new vertices are red. The outer face is always on the outside, and shadowed parts contain at least one vertex.

If we choose one special edge in the subgraph that is affected by the reduction operation — as indicated by the arrows in Figure 4.5 — each way to apply this reduction corresponds to an edge of P. We can choose an orbit of edges under the symmetry group of P by constructing a canonical labelling of the vertices (see Section 3) and choosing the orbit of the edge with the lowest numbered vertices. The canonical parent of P is then obtained by applying the corresponding reduction.

During the construction, we will try each possible extension in all possible ways, and then check if it is the inverse of the reduction used to get the canonical parent of the resulting predecoration. If that is the case, we can continue to extend this predecoration.

It is possible to construct all predecorations with fewer extensions,

but it is important that a canonical reduction always results in a valid predecoration. The order of extensions 1–4 ensures that a canonical reduction never increases n_A, and extensions 5–7 ensure that a canonical reduction never increases $n_A + n_B + n_C$. Extensions 8–10 are necessary when none of the other reductions are possible, so that each predecoration different from the base decorations has a possible reduction. We will prove this in Lemma 4.4 and Theorem 4.5.

Lemma 4.3. *An extension applied to a predecoration results in another predecoration if it keeps $n_A \leq 2$ and $n_A + n_B + n_C \leq 3$. Only extensions 1, 2 and 5 possibly violate this condition.*

Proof. It is easy to see that each extension can only create new inner faces that are quadrangles, and inner vertices with degree at least 3.

The only extensions that can increase n_A are extensions 1 and 2. The only extension that can increase n_B is extension 2. The only extension that can increase n_C is extension 5. □

This makes it easier to keep count of n_A, n_B and n_C during the construction.

Lemma 4.4. *Let P be a predecoration different from the base predecorations. By applying one of the reductions from Figure 4.5, P can be reduced to a graph containing fewer vertices or a graph containing the same number of vertices but fewer edges.*

Furthermore, if we apply the reduction with the smallest number among all possible reductions, the resulting graph is again a predecoration.

Proof. For the first part, it is clear that each reduction results in a 'smaller' graph, so we only need to verify that at least one reduction can be applied. If P contains at least one quadrangle, there is at least one quadrangle Q with an edge in the outer face. Since P is not C_4, there is at least one other vertex not contained in Q in the graph, and reduction 10 is possible. If there is no quadrangle in P, reduction 1 is possible.

For the second part, it is immediately clear that all reductions preserve the properties that all inner faces are quadrangles and that all inner

reduction	n_A	n_B	n_C
1		n_A	
2	1	1	n_B
3,4	1		
5,6,7	1	1	3/4
8	2/5	5	6/7
9	2/8	8	8
10	2/9	9	9

Table 4.1. Read this table as:

Reduction i can increase n_X,
but only if n_Y is decreased by the same amount.
Reduction i can increase n_X,
but only if reduction j/k can be applied too.

vertices have degree at least 3. It remains to be proven that for the new graph $n_A \leq 2$ and $n_A + n_B + n_C \leq 3$.

Some reductions can increase n_A, n_B or n_C, but only if another reduction with a smaller number can also be applied. This is the reason that we need so many extension operations in that particular order. In Section 2, all these situations are given.

It is impossible to increase n_A with a reduction that has the smallest possible number. Therefore, we still have $n_A \leq 2$ in the new graph.

Reduction 1 can increase n_B, but only by removing a vertex of degree 2 neighbouring a vertex of degree 1, i.e. by decreasing n_A by the same amount. Therefore, we still have $n_A + n_B + n_C \leq 3$ in the new graph.

Reduction 2 can increase n_C, but only by decreasing n_B by the same amount. Therefore, we still have $n_A + n_B + n_C \leq 3$ in the new graph. □

Theorem 4.5. *The algorithm described in Algorithm 1 generates all predecorations.*

Proof. This follows immediately from [McK98] and Lemma 4.4. □

Algorithm 1 Construction of predecorations.

```
function EXTEND(P)
    output P
    for i = 1, ..., 10 do
        for O an orbit of edges in the outer face of P do
            e ← edge in O
            P' ← apply extension i to edge e of P
            if P canonical parent of P' then
                EXTEND(P')
for G a base predecoration do
    EXTEND(P)
```

3 Construction of chamber decorations

Now that we can construct all predecorations, we can use the homomorphism principle [GLM97] and complete each predecoration in all possible ways to get all chamber decorations with Algorithm 2. We first have to compute the symmetry group of the predecoration, in order to avoid completions that result in the same chamber decoration. After the first 4 steps, all symmetry is broken by choosing v_0, v_1 and v_2.

We do not have to take isomorphisms into account, since two isomorphic chamber decorations will have isomorphic predecorations.

Note that it might not be possible to complete a predecoration in Step 6 such that there are no cut-vertices left.

3.1 Connectivity

In Step 7, we will always obtain a chamber decoration. The additional properties for 2-connected and 3-connected chamber decorations have to be checked. The properties in the outer face cannot be checked earlier in the construction process, because they depend on the chosen completion. But we can prevent type-1 cycles of length 2 and cycles of length 4 during the construction. It is clear that once a type-1 cycle is created during the construction, it cannot be destroyed later. So

Algorithm 2 Complete a predecoration in all possible ways.

1. If $n_A > 0$, label the corresponding vertices of degree 1 with v_0 or v_2 in all non-isomorphic ways.
2. If $n_B + n_C > 0$, label the corresponding vertices with v_0, v_1 or v_2 in all non-isomorphic ways.
3. If v_1 is not yet chosen, label an outer vertex with v_1 or add a new type-1 vertex v_1 of degree 2 in the outer face in all non-isomorphic ways.
4. If v_0 or v_2 is not yet chosen, label two outer vertices with v_0 and v_2 in all non-isomorphic ways.
5. Fill all inner quadrangles with a type-1 vertex.
6. Add type-1 vertices of degree 3 in the outer face in all possible ways, such that there are no cut-vertices or vertices of degree 2 left.
7. Check whether the result is a chamber decoration.

if we avoid the creation of type-1 cycle of length 2 or 4, we will still construct all the desired chamber decorations.

The only way to create a first type-1 cycle of length 2 is by applying extension 10 to a predecoration with an outer face of size 4. This can easily be avoided. The only way to create a non-empty type-1 cycle of length 4 is by applying extension 10 to a predecoration with an outer face of size 6. We can avoid this too.

To check the other properties after the completion, we can loop over the outer face of the chamber decoration, and mark all vertices one inner edge away from side i with i. If we encounter a vertex on side i that is marked with i, the chamber decoration is not 2-connected. If a vertex is marked two times with the same number, or a vertex on side 1 is marked with 0 or vice versa, the chamber decoration is not 3-connected.

3.2 Inflation factor

Although it is interesting to construct all possible chamber decorations, we are more interested in the chamber decorations with a given inflation factor. Unfortunately, we cannot determine the inflation factor before the predecoration is completed since chamber decorations with different inflation rates might have the same predecoration, but we can compute lower and upper bounds.

Given a predecoration P, for each chamber decoration that has P as its underlying predecoration, each quadrangle of P corresponds to 4 chambers and each cut-vertex of which the removal leaves $k \geq 2$ components requires $2(k-1)$ extra chambers. So

$$4 \cdot (\text{number of quadrangles}) + 2 \cdot \sum_{\text{cut-vertices}} (\text{occurrences in outer face} - 1)$$

is a lower bound for the inflation factor. The maximal inflation factor of a predecoration is reached by adding as much type-1 vertices as possible in the outer face. This will result in exactly one chamber for each edge in the outer face. In combination with the 4 chambers in each quadrangle, this results in 2 chambers (one at each side) for each edge of the predecoration. So the maximal inflation factor is

$$2 \cdot (\text{number of edges}).$$

If the lower bound for the inflation factor of a predecoration is already higher than the desired inflation factor, we do not have to extend it further as it can only increase. If the upper bound is lower than the desired inflation factor, we have to extend it, but we do not have to try to complete it.

4 Results

Using Algorithms 1 and 2, we implemented a computer program [Goe19] to generate all k-connected chamber decorations with a given inflation factor. On a 2.70 GHz Intel Core i7 processor, the program generates approximately 500 chamber decorations per millisecond. The results of this program are given in Table 4.2. The chamber decorations for inflation rates $r \leq 8$ are given in Table 4.3.

The two lsp operations with inflation factor 1 are obviously identity and dual. The lsp operations with inflation factor 2 are ambo and join, and the ones with inflation factor 3 are truncate, zip, needle and kis. Up to here, all lsp operations were already described by Conway [CBG08] or others. For the left chamber decoration with inflation factor 4, only two of the 4 related lsp operations (chamfer and subdivide) are already named. The first chamber decoration for which none of the related lsp operations (including dual and mirrored ones) are already named, is the 2-connected lsp operation with inflation factor 5. The first unnamed 3-connected lsp operations are the three leftmost chamber decorations with inflation factor 6.

These results are verified for inflation factor up to 23 by an independent implementation that constructs all triangulations, filters the chamber decorations out, applies them to a polyhedron, checks the connectivity and filters the isomorphic ones out.

r	k-connected decorations			pre-decorations
	$k=1$	$k=2$	$k=3$	
1	2	2	2	1
2	2	2	2	1
3	4	4	4	1
4	6	6	6	2
5	6	6	4	2
6	20	20	20	4
7	28	28	20	7
8	58	58	54	8
9	82	82	64	7
10	170	168	144	19
11	204	200	132	16
12	496	492	404	50
13	650	640	396	42
14	1432	1400	1112	118
15	1824	1786	1100	109
16	4114	3952	2958	298
17	5078	4900	2769	300
18	11874	11150	7972	749
19	14808	14058	7560	782
20	33978	30998	21300	1902
21	41794	38964	20076	2056
22	97096	85976	56296	4893
23	118572	107784	52380	5419
24	277208	237482	148956	12615
25	337216	298546	138384	14153

Table 4.2. The number of k-connected chamber decorations up to inflation factor $r = 50$. The number of predecorations that can be completed to a chamber decoration with given inflation factor are given too. Some of these predecorations cannot be completed to 2-connected or 3-connected chamber decorations, and can be avoided during the construction of necessary.

r	k-connected decorations			pre-decorations
	$k=1$	$k=2$	$k=3$	
26	788342	652236	392096	32665
27	953060	820960	362499	36953
28	2239396	1786222	1027488	84853
29	2697088	2250816	945612	96491
30	6350014	4875076	2687408	220646
31	7618068	6153604	2466156	251104
32	17972390	13262574	7007118	573547
33	21487746	16773086	6409664	654663
34	50805716	35985748	18222032	1491540
35	60573248	45592594	16623268	1706755
36	143425040	97394726	47287986	3878836
37	170530518	123628298	43038260	4446426
38	404413576	262983002	122451618	10085305
39	479711448	334473144	111200316	11582891
40	1139138344	708583784	316474370	26222191
41	1348351130	902941632	286833388	30171977
42	3205480518	1905439964	816499872	68173558
43	3786676452	2432756832	738710312	78555301
44	9012067074	5114410850	2103076126	177209924
45	10627114786	6542306684	1899585208	204462214
46	25315977970	13703891826	5408673658	460610114
47	29803635416	17563047708	4878021416	531956837
48	71060710972	36659785890	13890262050	1196981883
49	83530248078	47071210950	12510181242	1383455418
50	199322142978	97920815584	35624781874	3110039163

Table 4.2. (continued)

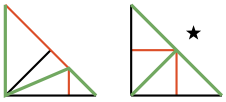
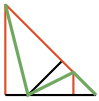
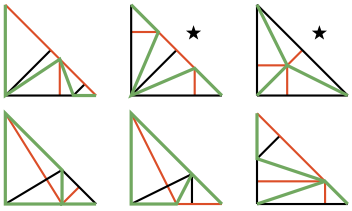
inflation factor	$k = 2$	$k = 3$
1		
2		
3		
4		
5		
6		

Table 4.3. All chamber decorations with inflation factor up to 8. The green lines are edges of type 1. The black lines are edges of type 0 and 2. For each of the given chamber decorations, the edges of type 0 and 2 can be chosen in two different ways. All chamber decorations except the symmetric ones (marked with a star) can be mirrored. So each starred chamber decoration represents two related lsp operations, and the unstarred ones represent four related lsp operations.

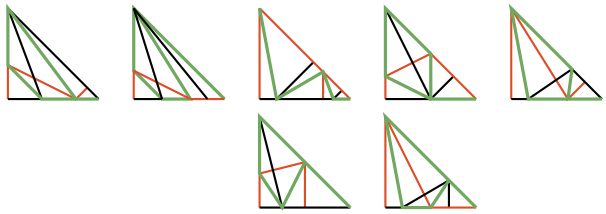
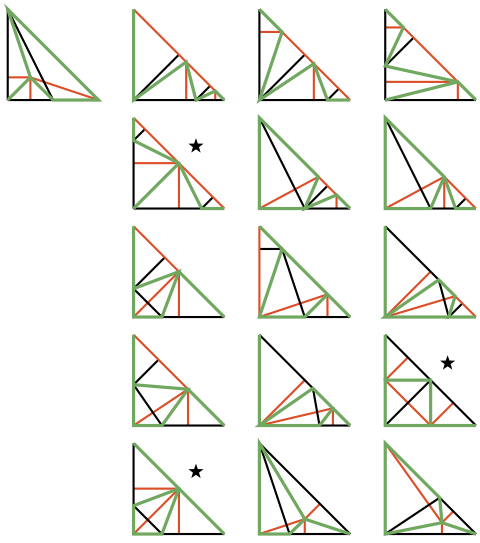
inflation factor	$k = 2$	$k = 3$
7		
8		

Table 4.3. (continued)

References

[CBG08] J.H. Conway, H. Burgiel and C. Goodman-Strauss. *The Symmetries of Things*. A K Peters, 2008.

[Goe19] P. Goetschalckx. *decogen*. 2019. URL: https://github.com/314eter/decogen.

[GLM97] T. Grüner, R. Laue and M. Meringer. "Algorithms for group actions: homomorphism principle and orderly generation applied to graphs". In: *Groups and Computation II*. Ed. by L. Finkelstein and W. Kantor. DIMACS Series in Discrete Mathematics and Theoretical Computer Science 28. 1997, pp. 113–122.

[McK98] B.D. McKay. "Isomorph-Free Exhaustive Generation". In: *Journal of Algorithms* 26.2 (1998), pp. 306–324.

5

Local orientation-preserving symmetry preserving operations

Most Goldberg operations and some of the extended Conway operations — like snub (see Figure 5.1), gyro, propeller, etc. — are chiral, so they only preserve orientation-preserving symmetries. In order to also cover these, we can generalize lsp operations by decorating double chambers instead of single chambers, similar to what Goldberg did in [Gol37] for Goldberg operations. We call these local orientation-preserving symmetry preserving (*lopsp*) operations. In this chapter, we formalize this approach.

We will prove that each lopsp operation can be represented by a *double chamber patch*, but in contrast to the chamber decoration of an lsp operation, this one is not necessarily unique. We introduce the *double chamber decoration* of an lopsp operation, which can be easily constructed from the double chamber patch but is independent of the chosen patch, and therefore unique for each lopsp operation. After some auxiliary results, we prove that the double chamber decoration is an invariant for equivalent lopsp operations. This makes it possible to identify an lopsp operation with its double chamber dec-

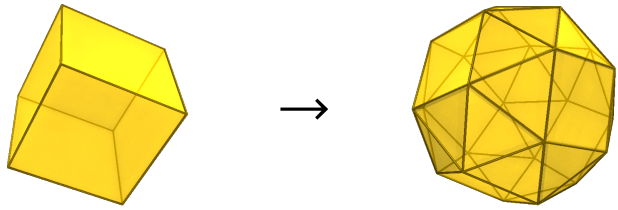


Figure 5.1. The Conway operation snub applied to the cube, resulting in the Archimedean solid called *snub cube*.

oration. Finally, we give a combinatorial characterization of double chamber decorations independent of the corresponding lopsp operation, and identify the double chamber decorations of *2-connected* and *3-connected* lopsp operations. Such a characterization is one of the first steps towards constructing a generation algorithm in Chapter 6. Finally, we prove that 2-connected resp. 3-connected lopsp operations preserve 2-connectivity resp. 3-connectivity, which makes it possible to see 3-connected lopsp operations as operations on polyhedra.

1 Chiral operations

An example of the construction of a chiral Goldberg operation is given in Figure 5.2. A quadrangular *double chamber patch* v_1, v_0, v_2, v_0' consisting of the triangles v_1, v_0, v_2 and its counterpart v_1, v_0', v_2 is cut out of the hexagonal tiling H. Given a plane graph G with chamber system C_G, we can glue this patch into each double chamber of C_G. The result is a plane graph G' with the same orientation-preserving symmetries as G, but not necessarily the same orientation-reversing symmetries.

The symmetries of the hexagonal tiling ensure that after cutting and gluing the patch everything still fits together. But in order to permit a combinatorial approach to the operations, we prefer to cut over a simple path P in C_H instead of cutting through edges and faces in arbitrary places. We will prove in Lemma 5.4 that it is always possible to find such a path.

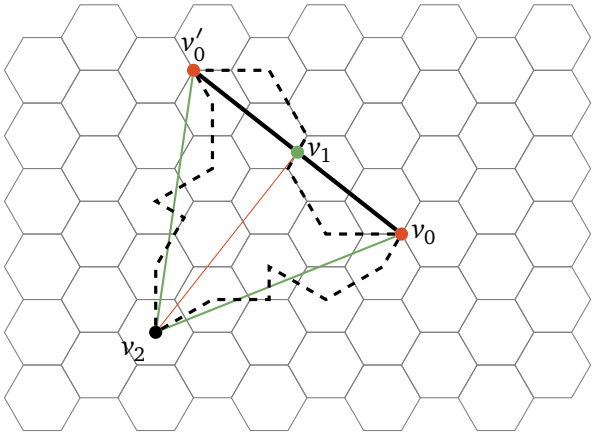


Figure 5.2. The double chamber patch of the Goldberg $(5,3)$ operation, with a simple path P in dashed lines.

It would be easy if we could split the double chamber patch into two separate triangles such that each triangle corresponds to the chamber decoration of an lsp operation, and decorate each of the two types of chambers of C_G with one of these two patches. Unfortunately, this is not always possible. In Figure 5.3, such an example is given. This is a double chamber patch for the lopsp operations snub.

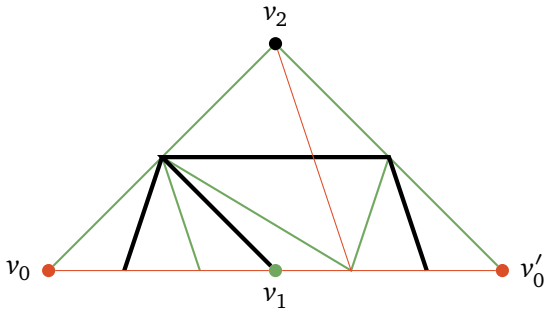


Figure 5.3. A double chamber patch for snub (see Figure 5.1). There is no path between v_1 and v_2 that splits the double chamber patch into two triangular single chamber patches.

2 Lopsp operations

We define lopsp operations in a similar way to lsp operations, but instead of decorating each chamber with a single chamber decoration, we will decorate double chambers.

Definition 5.1. Let T be a periodic tiling embedded in the plane, with points v_0 and v_2 such that v_0 is the center of a rotation ρ_{v_0} by 120 degrees in clockwise direction that is a symmetry of T and v_2 is the center of a rotation ρ_{v_2} by 60 degrees in clockwise direction that is a symmetry of T.

We call (T, v_0, v_2) a *local orientation-preserving symmetry preserving operation*, lopsp operation for short.

Let $v_0' = \rho_{v_2}(v_0)$. The rotation $\rho_{v_2} \circ \rho_{v_0}$ is a rotation by 180 degrees. We call the center of this rotation v_1, and the rotation ρ_{v_1}.

In contrast to lsp operations, there is no obvious way to apply lopsp operations. We want to cut out the double chamber patch v_2, v_0, v_1, v_0' and glue it into each double chamber, but the straight lines between these vertices do not always coincide with edges of C_T, and if we allow other cut-paths there are multiple possibilities (see Figure 5.4).

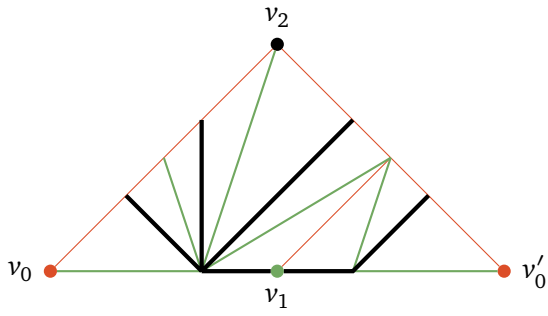


Figure 5.4. Another double chamber patch for snub (see Figure 5.1).

No matter how we cut out this patch, the result after gluing the copies of the patch together will be the same. If we choose another path between v_1 and v_0 or between v_0 and v_2, we have to adapt the path between v_1 and v_0' resp. v_0' and v_2 accordingly, and the changes will

cancel each other out when we glue the patches together. This suggests that if we identify the vertices and edges on the border v_1, v_0, v_2 of the patch with the vertices and edges on the border v_1, v_0', v_2, the result is a triangulation of the sphere invariant under the chosen path. In Figure 5.5 the resulting triangulation for the snub operation is given. We can even construct this triangulation without choosing a path.

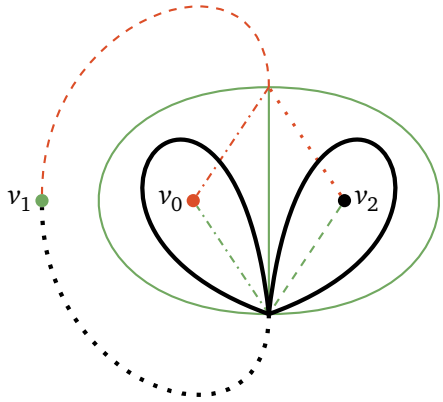


Figure 5.5. The double chamber decoration of snub (see Figure 5.1), with the simple path corresponding to the double chamber patches of Figures 5.3 and 5.4 in dashed resp. dotted lines.

For $C_T = (E, \sigma, \theta)$, consider the quotient set $\overline{E} = E/\langle \rho_{v_0}, \rho_{v_2} \rangle$. With $\overline{e}$ the equivalence class of e in $\overline{E}$, we define

$$\begin{aligned} \overline{\sigma}(\overline{e}) &= \overline{\sigma(e)}, \\ \overline{\theta}(\overline{e}) &= \overline{\theta(e)}, \\ \overline{t}(\overline{e}) &= t(e). \end{aligned}$$

Definition 5.2. The plane graph $(\overline{E}, \overline{\sigma}, \overline{\theta})$ described above together with labelling function $\overline{t}\colon \overline{E} \to \{0,1,2\}$ and special vertices v_0, v_1 and v_2 is called the *double chamber decoration of the lopsp operation* (T, v_0, v_2).

Since for all $e_1, e_2 \in E$ with $\overline{e_1} = \overline{e_2}$ there exists a symmetry $\rho \in \langle \rho_{v_0}, \rho_{v_2} \rangle$ with $\rho(\sigma(e)) = \sigma(\rho(e))$, $\rho(\theta(e)) = \theta(\rho(e))$ and $\rho(t(e)) = t(\rho(e))$ such that $\rho(e_1) = e_2$, it is easy to prove that $\overline{\sigma}$, $\overline{\theta}$ and $\overline{t}$ are well-defined and $(\overline{E}, \overline{\sigma}, \overline{\theta})$ is indeed a plane graph. Although we

defined the labelling function t only on edges, the types of vertices can be easily derived from the types of its incident edges.

Lemma 5.3. *The double chamber decoration of an lopsp operation is a plane triangulation.*

Proof. Since C_T is a triangulation, we know that $(\sigma\theta)^3(e) = e$ for all edges $e \in E$. It follows immediately that $(\overline{\sigma}\overline{\theta})^3(\overline{e}) = \overline{e}$ for all edges $\overline{e} \in \overline{E}$. Since

$$\overline{t}(\overline{\sigma}\overline{\theta}(\overline{e})) = t(\sigma\theta(e)) \neq t(\theta(e)) = t(e) = \overline{t}(\overline{e}),$$

it is impossible that $\overline{\sigma}\overline{\theta}(\overline{e}) = \overline{e}$. Therefore, the size of each orbit of $\langle\overline{\sigma}\overline{\theta}\rangle$ is 3, which means that all the faces are triangles. □

Now that we obtained the double chamber decoration D without choosing a cut-path, we can choose a path in D instead of C_T. This is easier to do, because we do not have to take the symmetries into account. We can always find a path along the edges of C_T, without crossing through edges or faces.

Lemma 5.4. *If D is the double chamber decoration of an lopsp operation, there exists a simple path P between v_1 and v_2 through v_0.*

Proof. Since D is a plane triangulation with at least 4 vertices, it is 3-connected and therefore also 2-connected. It stays 2-connected if we temporarily add a vertex w with edges to v_1 and v_2. By Menger's theorem [Men27], there exist two disjoint paths between w and v_0. This is only possible if there are disjoint paths from v_1 to v_0 and from v_0 to v_2. □

We apply a double chamber decoration D to a plane graph G by cutting D open along the simple path P from the lemma above, which is the subdivided patch v_1, v_0, v_2, v_0' that we glue into each double chamber of G. Instead of cutting and gluing, we can describe this application combinatorially.

Denote the set of directed edges on the path from v_2 to v_0 by P_2, and their inverses by P_2'. Denote the set of directed edges on the path from v_1 to v_0 by P_1, and their inverses by P_1'.

There is a one-to-one correspondence between the directed edges of a plane graph $G = (E, \sigma, \theta)$ and the double chambers of C_G, where each edge e corresponds to the double chamber c_e immediately to its left. The operations $s_1(c_e) = c_{\theta(e)}$ and $s_2(c_e) = c_{\sigma^{-1}\theta(e)}$ correspond to traversing the cyclic order around vertices of type 1 resp. 2. We call the set of double chambers of G along with s_1 and s_2 the *double chamber system* of G.

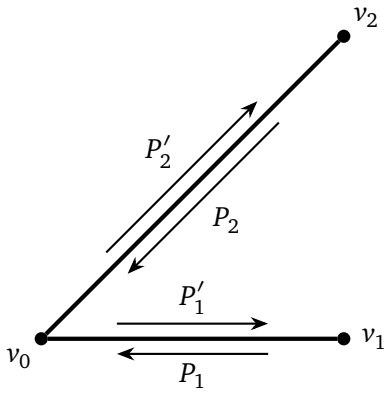


Figure 5.6. A double chamber decoration with simple path P.

Definition 5.5. Given a plane graph G with double chamber system C and a double chamber decoration $D = (E, \sigma, \theta)$ with simple path P satisfying Lemma 5.4, the *application of* (D, P) *to* G results in a plane graph $D_P(G) = (E \times C, \sigma_P, \theta_P)$ with $\sigma_P((e, c)) = (\sigma(e), s_{P,e}(c))$ and $\theta_P((e, c)) = (\theta(e), s_{P,e}(c))$ where

$$s_{P,e} = \begin{cases} s_2 & \text{if } e \in P_2 \\ s_1^{-1} & \text{if } e \in P_1 \\ s_2^{-1} & \text{if } e \in P_2' \\ s_1 & \text{if } e \in P_1' \\ \mathbf{1} & \text{else} \end{cases}$$

It is possible that there is more than one simple path that satisfies Lemma 5.4. We still have to prove that the result of the operation does not depend on the chosen path P. We will do that in Theorem 5.7, but we first introduce some new terminology.

Given a double chamber decoration D with two simple paths P and Q satisfying Lemma 5.4, consider the subgraph of C_D consisting of all the edges in P and Q (see Figure 5.7 for an example). In order to avoid confusion, we will refer to the faces of this subgraph as *regions*. With each directed edge e of C_D we associate exactly one region R_e. If e is an edge in P or Q we choose the region at the left-hand side of e, and for all other edges we choose the containing region. A *region path* $R_0, \ldots, R_n$ is a sequence of regions such that for each $i < n$ there exists an edge $e_i \in Q \setminus P$ such that e_i is associated with R_i and $\theta(e_i)$ is associated with R_{i+1}. A region path corresponds to the operation $r_1 \cdots r_n$ with $r_i = s_{Q,e_i}$. Two region paths are called equivalent if they correspond to the same operation.

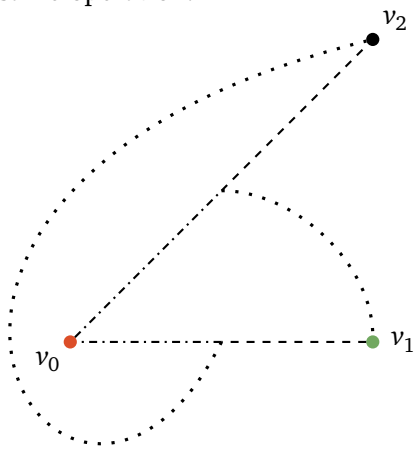


Figure 5.7. The regions of the double chamber decoration of snub (see Figure 5.5) with two simple paths in dotted and dashed lines.

Lemma 5.6. *Given a double chamber decoration D with two simple paths P and Q satisfying Lemma 5.4, there exists a region $R_{P,Q}$ such that there is a region path between $R_{P,Q}$ and a region incident to v_2 with an associated operation of the form s_2^k, and a region path between $R_{P,Q}$ and a region incident to v_1 with an associated operation of the form s_1^l.*

Proof. Choose a region path $R = R_0, \ldots, R_n$ with R_0 incident to v_1 and R_n incident to v_2 and associated operation $r = r_1 \cdots r_n$. Such a region path exists because P contains no cycles, so all regions are connected. If we add one vertex in each region and one vertex on each edge in Q,

and an edge between a vertex in a region and a vertex on an edge if the edge is in the border of the region, this region path induces a path on these edges in a canonical way, and each operation r_i corresponds to an intersection of R and Q. An example is given in Section 2.

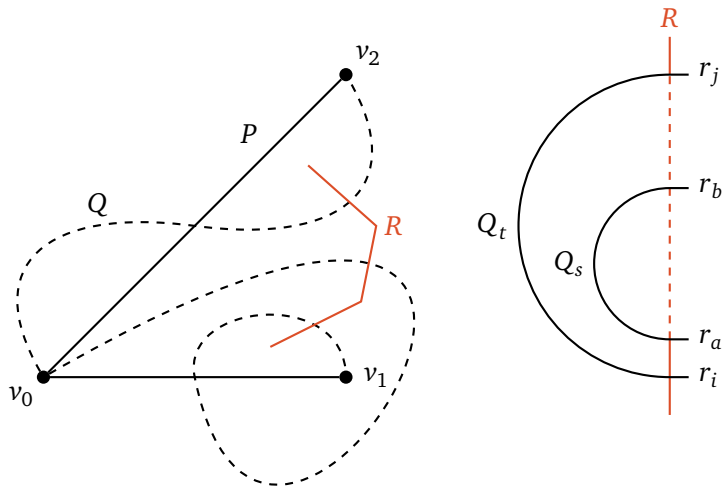


Figure 5.8. Examples of region paths. The second figure shows a region path that intersects with Q multiple times.

We will prove that if r_i and r_j correspond to two intersections of R and Q_t with $t \in \{1,2\}$ that are consecutive on Q_t, then $r_{i+1} \cdots r_{j-1}$ is the identity operation. For $j = i+1$ this is obvious. Suppose $j > i+1$. The subpath of Q_t between r_i and r_j together with the region path between r_i and r_j forms a closed cycle. If there is an intersection with Q_s in r_a with $a = i+1$, there will be another intersection in r_b with $a < b < j$ and $r_b = r_a^{-1}$, as illustrated in Section 2. We can assume by induction that $r_a \cdots r_b = r_a r_b = \mathbf{1}$. If $b < j-1$, we can repeat this for $a = b+1$ until $b = i+1$ and thus $r_{i+1} \cdots r_{j-1} = \mathbf{1}$.

Take m so that $r_m \in \{s_1, s_1^{-1}\}$ and $r_i \in \{s_2, s_2^{-1}\}$ for all $i > m$. If Q_2 crosses R in r_a with $1 < a < m$ and $r_i \in \{s_1, s_1^{-1}\}$ for $i < a$, it will cross again in r_b with $a < b < m$, and $r_a \cdots r_b = \mathbf{1}$. We can repeat this as long as there is an $r_c \in \{s_2, s_2^{-1}\}$ with $b < c < m$, until $r_1 \cdots r_m = s_1^k$.

Since $r_{m+1} \cdots r_n = s_2^l$, the region between r_m and r_{m+1} satisfies the conditions of $R_{P,Q}$. □

We are now ready to prove that the application of a double chamber decoration $D = (E, \sigma, \theta)$ to a graph G with double chamber system C is independent of the chosen simple path P. In order to do that, we will construct an isomorphism between the plane graphs $D_P(G)$ and $D_Q(G)$, with Q another simple path satisfying Lemma 5.4. By choosing the region $R_{P,Q}$, we fix canonical points $R_{P,Q} \times C$ that will be invariant under this isomorphism.

Theorem 5.7. *Given a plane graph G with double chamber system C and a double chamber decoration $D = (E, \sigma, \theta)$ with two simple paths P and Q satisfying Lemma 5.4, there exists an isomorphism between $D_P(G)$ and $D_Q(G)$.*

Proof. Choose a region $R_{P,Q}$ satisfying Lemma 5.6 and consider the function

$$f : E \times C \to E \times C : (e, c) \mapsto (e, s_{P,Q,e}(c))$$

with $s_{P,Q,e}$ the operation associated with a region path from $R_{P,Q}$ to R_e. We will first prove that f is a homomorphism between $D_P(G)$ and $D_Q(G)$. Since

$$\begin{aligned}
\sigma(f((e,c))) &= \sigma((e, s_{P,Q,e}(c))) = (\sigma(e), s_{Q,e} s_{P,Q,e}(c)) \\
f(\sigma((e,c))) &= f((\sigma(e), s_{P,e}(c))) = (\sigma(e), s_{P,Q,\sigma(e)} s_{P,e}(c)),
\end{aligned}$$

$$\begin{aligned}
\theta(f((e,c))) &= \theta((e, s_{P,Q,e}(c))) = (\theta(e), s_{Q,e} s_{P,Q,e}(c)) \\
f(\theta((e,c))) &= f((\theta(e), s_{P,e}(c))) = (\theta(e), s_{P,Q,\theta(e)} s_{P,e}(c)),
\end{aligned}$$

we only have to prove that $s_{Q,e} s_{P,Q,e} = s_{P,Q,\sigma(e)} s_{P,e}$.

Consider the case $s_{P,e} = \mathbf{1}$. The operation $s_{P,Q,\sigma(e)}$, corresponding to a region path from $R_{P,Q}$ to $R_{\sigma(e)}$, is equal to $s_{P,Q,e}$ followed by the operation corresponding to the region path crossing e, which is $s_{Q,e}$. Therefore, $s_{P,Q,\sigma(e)} s_{P,e} = s_{Q,e} s_{P,Q,e}$.

If $s_{P,e} = s_1$, there is a region path from $R_{P,Q}$ to R_e consisting of a region path from $R_{P,Q}$ to a region R_1 incident to v_1, corresponding to operation s_1^k, followed by a region path from R_1 to R_e, corresponding

to operation r. Since $e \in P_1'$, the region path from R_1 to R_e can follow the left-hand side of P_1. The region path from $R_{P,Q}$ to $\sigma(e)$ starts with the same region path to R_1. We can now follow the region path along the right-hand side of P_1 to $R_{\sigma(e)}$, corresponding to operation r', after we go around v_1 which corresponds to operation s_1^{-1}. In Figure 5.9, we see that r' is equal to r followed by $s_{Q,e}$. Therefore,

$$s_{P,Q,\sigma(e)} s_{P,e} = r' s_1^{-1} s_1^k s_1 = s_{Q,e} r s_1^k = s_{Q,e} s_{P,Q,e}.$$

For $s_{P,e}$ equal to s_1^{-1}, s_2 and s_2^{-1}, the proof is similar.

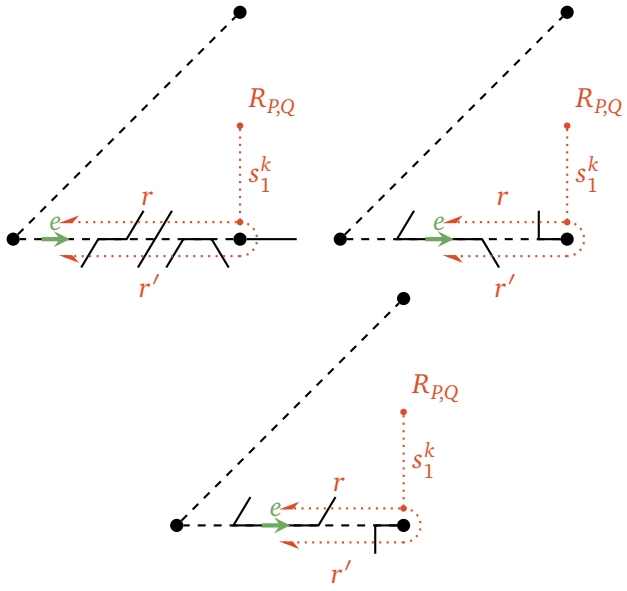


Figure 5.9. Some examples for $s_{P,e} = s_1$.

Suppose $f((e,c)) = f((e',c'))$, i.e. $(e, s_{P,Q,e}(c)) = (e', s_{P,Q,e'}(c'))$. It follows immediately that $e = e'$, and since $s_{P,Q,e}$ is a permutation we have that $c = c'$. Thus $(e,c) = (e',c')$ and f is injective. For each $(e,c) \in E \times C$, $f((e, s_{P,Q,e}^{-1}(c))) = (e,c)$, and thus f is surjective.

Since f is a bijective homomorphism, it is an isomorphism between $D_P(G)$ and $D_Q(G)$. □

3 Double chamber decorations

In the previous section we constructed the double chamber decoration for a given lopsp operation. This double chamber decoration contains all the necessary information in order to apply the decoration to an embedded graph, but does not depend on the tiling T or the simple path P chosen to define and apply the lopsp operation. Since two lopsp operations are equivalent if and only if they have the same double chamber decoration, it is easier to work with the double chamber decorations directly instead of deriving them from lopsp operations. But in order to do that, we need a full characterization of these graphs. This is similar to what we did for lsp operations in Theorem 3.3.

Theorem 5.8. *A plane triangulation D with vertex set V and edge set E, together with a labelling function $t : V \cup E \to \{0,1,2\}$ and three special vertices v_0, v_1, v_2 is a double chamber decoration of an lopsp operation if and only if*

1. *for each edge $e = (v,w)$, $\{t(e), t(v), t(w)\} = \{0,1,2\}$;*
2. *for each vertex v with $t(v) = i$, the types of the edges incident to v are alternating between j and k with $\{i,j,k\} = \{0,1,2\}$;*
3. *for each vertex v different from v_0, v_1, v_2*

$$\begin{aligned} t(v) = 1 \quad &\Rightarrow \quad \deg(v) = 4 \\ t(v) \neq 1 \quad &\Rightarrow \quad \deg(v) > 4 \end{aligned}$$

and

$$\begin{aligned} t(v_0), t(v_2) &\neq 1 \\ \deg(v_0), \deg(v_2) &\geq 2 \\ t(v_1) = 1 \quad &\Rightarrow \quad \deg(v_1) = 2 \\ t(v_1) \neq 1 \quad &\Rightarrow \quad \deg(v_1) \geq 4. \end{aligned}$$

Proof. It is easy to verify that the double chamber decoration of an lopsp operation satisfies these properties.

Given a graph D that satisfies the properties, there exists a simple path P between v_1 and v_2 through v_0, since the proof of Lemma 5.4 holds

for all plane triangulations. We can cut D open along this path to get a subdivided patch D', and glue this patch into each double chamber of the hexagonal tiling H. The result will be a chamber system C_T of a tiling T.

We will now prove that the type-2 subgraph of D', consisting of all type-2 edges, is connected. Let u and v be two vertices in the type-2 subgraph. Since every face of D' is a cycle, D' is 2-connected. Menger's theorem [Men27] gives us that there exist two vertex-disjoint paths between u and v. Since all faces of D' except for the outer face are triangles, these two paths form a cycle with only triangles on the inside. Since u is in the type-2 subgraph, it has type 0 or 1, and there is an edge (u, u') of type 2 on or in the cycle. If $u' \neq v$, we can do the same for vertices u' and v, and we can choose a cycle that contains fewer triangles than the previous one. By induction, there exists a path between u and v in the type-2 subgraph of D'.

Given vertices u and v in the type-2 subgraph of C_T, there exists a sequence of chambers $C_0, \dots, C_n$ of H such that two consecutive chambers C_i and C_{i+1} share one side, and u is contained in C_0 and v in C_n. Since there are at least two vertices on each side of D', and they are not both of type 2, at least one of them is in the type-2 subgraph of C_T. Thus, there is a type-2 path between u and v that passes through all chambers in the sequence $C_0, \dots, C_n$, and the type-2 subgraph of C_T is connected. It follows immediately that T is connected too.

We can choose the vertices of one double chamber of C_H in T as v_0, v_1, v_0' and v_2. Now (T, v_0, v_2) satisfies Definition 5.1 of an lopsp operation, and the double chamber decoration of this lopsp operation is D. □

We call an lopsp operation and the corresponding double chamber decoration k-connected if it is derived from a k-connected tiling T. For the following results, we need Lemma 2.18.

Theorem 5.9. *If G is a k-connected plane graph with $k \in \{1, 2, 3\}$ and O is a k-connected lopsp operation, then $O(G)$ is a k-connected plane graph.*

Proof. Suppose O is derived from a tiling T as in Definition 5.1.

For $k = 1$, we know that T and G are connected, and it follows easily that $O(G)$ is connected too.

For $k = 2$, we will prove that $O(G)$ is 2-connected. A type-1 cycle of length 2 in $C_{O(G)}$ is either completely contained in an area that was one double chamber of C_G before it was subdivided by O, or it is split between two areas of adjacent double chambers. Both cases cannot appear, as for any double chamber (resp. any pair of adjacent double chambers) of C_G there is an isomorphism between the area of this double chamber (resp. two double chambers) in $C_{O(G)}$ and the corresponding area in T, and T is connected and thus has no type-1 cycles of length 2. This implies that $C_{O(G)}$ contains no type-1 cycles of length 2.

For $k = 3$, we will prove that $O(G)$ is 3-connected. Consider a type-1 cycle C with edges $e_1, \ldots, e_n$ in $C_{O(G)}$. Let $C_1, \ldots, C_n$ be a sequence of double chambers of C_G such that e_i is contained in the area of C_i for $1 \leq i \leq n$. If $C_i = C_{i+1}$, we can remove C_i from the sequence. This results in the reduced sequence $C_1, \ldots, C_m$.

If C has length 2, then $m \leq 2$. Thus, the cycle is contained in one or two neighbouring areas, and it should be present in the tiling T too, which is impossible.

If C has length 4, then $m \leq 4$. Thus, the cycle is contained in the areas of at most 4 double chambers of C_G, and each double chamber has at least one vertex or edge in common with the previous and next one, but not the same for both of them. We will now construct a type-1 cycle in C_G though these chambers. Depending on the position of the common elements in each double chamber, we choose type-1 edges of C_G as in Figure 5.10.

This results in at most 4 edges that form a type-1 cycle or single edge C' in C_G. In Section 3 we choose two edges, but since v_0 and v_0' are of the same type, the path between them on the cycle C has to be at least of length 2 too. If C' is a type-1 cycle, it has to be empty since G is 3-connected. Thus, the situation is as in Section 3.

The type-1 cycle C in $C_{O(G)}$ is completely contained in the areas of double chambers of C_G adjacent to C'. The only situation where this would not necessarily imply a type-1 cycle in C_T is when C is a cycle

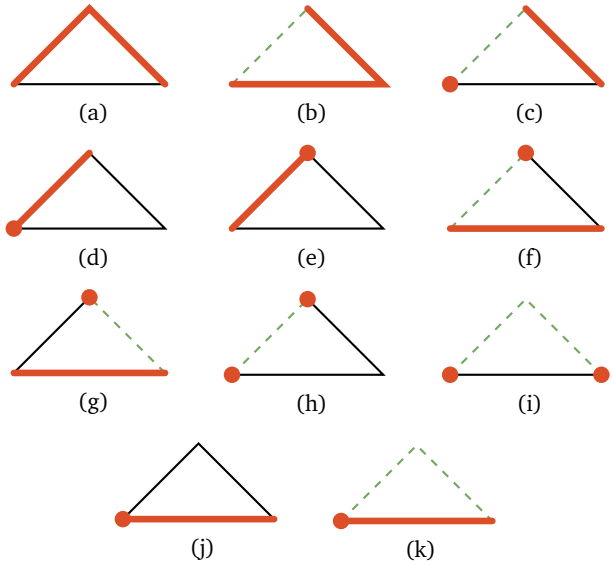


Figure 5.10. The thick red vertices and edges are the ones in common with the previous and next double chambers. The dashed green edges are part of the chosen cycle. The choice between (e) or (f) and (i) or (j) depends on the choice in the double chamber on the other side of the thick red edge (the cycle has to be connected).

of length 4 surrounding a type-2 vertex. This implies that C passes though 3 or 4 areas corresponding to double chambers of C_G, as illustrated in Section 3. There are at least two areas that contain only one edge of C. But since all the areas are isomorphic, it is easy to see that this is impossible. □

This theorem is particularly interesting for $k = 3$, for which it says that 3-connected lopsp operations are operations on polyhedra.

We can characterize the 2-connected and 3-connected double chamber decorations independently of the tiling.

Lemma 5.10. *A double chamber decoration D is 2-connected if and only if it has no type-1 cycle of length 2 with v_0, v_1 and v_2 on the same side.*

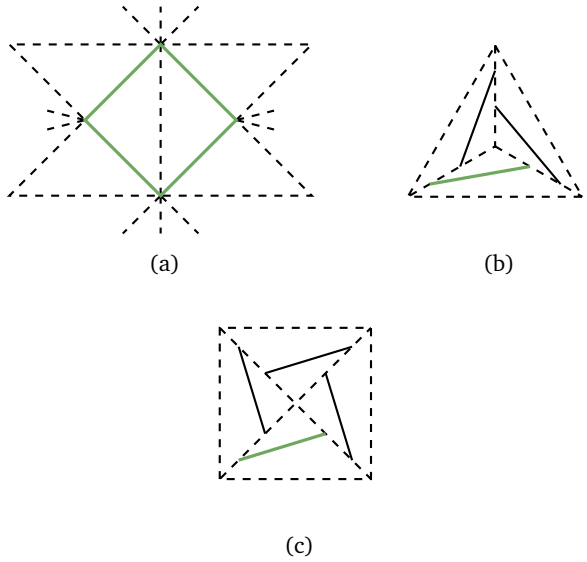
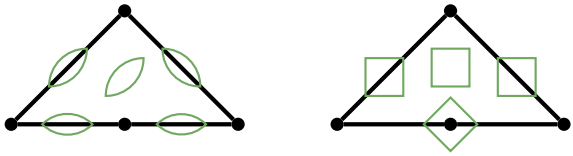


Figure 5.11. The type-1 cycle C' with adjacent double chambers.

Proof. Suppose D is not 2-connected. Then there is an lopsp operation (T, v_0, v_2) corresponding to D such that T is not 2-connected, so C_T has a type-1 cycle of length 2. According to Lemma 5.4, there exists a simple path between v_1 and v_2 through v_0 in D. This path corresponds to a simple cycle C through v_1, v_0, v_2, v_0' in T. The type-1 cycle of length 2 is either completely contained in C, or its two vertices are on the same section of C, as in Section 3. Both these situations imply the existence of a type-1 cycle in D that does not contain v_0, v_1 or v_2.

Suppose D has a type-1 cycle of length 2 with v_0, v_1 and v_2 on the same side. Given a simple path P between v_1 and v_2 through v_0, this cycle either contains no edge of P on its inside, or its two vertices are in P. Both these situations imply a type-1 cycle of length 2 in T. □

Lemma 5.11. *A double chamber decoration D is 3-connected if and only if it is 2-connected and has no non-empty type-1 cycle of length 4 with v_0, v_1 and v_2 on the same side or a type-1 cycle of length 2 with v_0 and v_2 on the same side.*



(a) (b)

Figure 5.12. The different situations where type-1 cycles of length 2 and length 4 can occur.

Proof. Suppose D is 2-connected but not 3-connected. Then there is an lopsp operation (T, v_0, v_2) corresponding to D such that C_T has a non-empty type-1 cycle of length 4. The possible positions for this cycle are given in Section 3, and all these situations imply the existence of a type-1 cycle of length 4 in D that does not contain v_0, v_1 or v_2 or a type-1 cycle of length 2 that contains only v_1.

Suppose D has a non-empty type-1 cycle of length 4 with v_0, v_1 and v_2 on the same side. Given a simple path P between v_1 and v_2 through v_0, this cycle either contains no edge of P on its inside, or it has two vertices in P. Both these situations imply a type-1 cycle of length 4 in T.

If D has a type-1 cycle with v_0 and v_2 on the same side, v_1 is on the other side or D is not 2-connected. This implies a 4-cycle in T that contains v_1. □

4 Composition

The algebraic structure of lopsp operations is very similar to lsp operations. Since each lsp operation is in fact an lopsp operation, we will see that the lsp operations are a *submonoid* of the lopsp operations.

Definition 5.12. Given two lopsp operations o_1 and o_2 derived from tilings T_1 and T_2, we can apply the operation o_2 to T_1. This will result in a connected tiling T', and since symmetries are preserved the points v_0 and v_2 will have the same rotation symmetry, and the line L_1 will

have the same mirror symmetry as in T_1. We call the lopsp operation $o_2o_1 = (T', v_0, v_2)$ the *composition* of o_1 and o_2.

The double chamber decoration of the composition o_2o_1 can be obtained by "applying" o_2 to the double chamber decoration of o_1. Note that a double chamber decoration is not always a valid chamber system since it can have vertices of degree 2, but it is still possible to apply another lopsp operation to the "double chambers".

Theorem 5.13. *The set of lopsp operations together with the composition operation forms a monoid, of which the lsp operations are a submonoid. They define a monoid action on the embedded graphs.*

Proof. The proof that the lopsp operations are a monoid and that they define an action is similar to the proof of Theorem 3.9.

It is obvious that an lsp operation $o = (T, v_0, v_2)$ satisfies the conditions of Definition 5.1. Since the mirror axes through v_0, v_1 and v_2 follow the edges of the barycentric subdivision, we can choose a simple path between v_1 and v_2 though v_0 such that it corresponds to the straight lines between v_0, v_1, v_0' and v_2. So there is a double chamber patch that obviously corresponds with two singe chamber decorations, and the result of applying the lsp operation or the lopsp operation is the same.

We already know that the lsp operations are closed under the composition operation, and they contain the identity element. So the lsp operations are a submonoid of the lopsp operations. □

Note that the composition of an lsp operation with a chiral lopsp operation is always chiral, and therefore not an lsp operation.

We can recognize the achiral lopsp operations by their double chamber decorations. If the double chamber decoration of an lopsp operation o has an orientation-reversing automorphism that maps v_0, v_1 and v_2 onto themselves, then o is achiral, and therefore an lsp operation.

Just as for lsp operations, each lopsp operation x has a number of related lopsp operations. The dual operation dx can be obtained by interchanging types 0 and 2 in the double chamber decoration. The

operation xd can be obtained by interchanging v_0 and v_2. Another related lopsp operation is the chiral pair, which can be obtained by reversing the orientation of the double chamber decoration.

5 Inflation

We use the same definition for inflation factor:

Definition 5.14. The *inflation factor* $r(o)$ of an lopsp operation o is the number of edges of $o(G)$ divided by the number of edges of G, for any plane graph G.

Since the double chamber decoration of an lsp operation o contains twice as many chambers as the single chamber decoration, the inflation factor $r(o)$ is half the number of chambers in the double chamber decoration. Each double chamber of an embedded graph G is replaces by the $2r(o)$ chambers of a double chamber patch in $o(G)$, so the number of chambers — and consequently the number of edges — increases with a factor $r(o)$.

The proofs of the following lemmas are similar to the proofs of Theorem 3.13 and Lemma 3.14.

Theorem 5.15. *The inflation factor $r(o)$ of an lopsp operation o is a representation of the monoid of lopsp operations.*

Lemma 5.16. *The only lopsp operations with inflation factor 1 are identity and dual. Consequently, they are the only invertible lopsp operations.*

Suppose D_o is the double chamber decoration with $2r$ chambers of an lopsp operation o. Take d_i the number of vertices of type i different from v_0, v_1 and v_2. Take $c_{i,j} = 1$ if v_j has type i. Then the matrix

$$M_o = \begin{pmatrix} r & 0 & 0 & 0 \\ d_0 & c_{0,0} & c_{0,1} & c_{0,2} \\ d_1 & c_{1,0} & c_{1,1} & c_{1,2} \\ d_2 & c_{2,0} & c_{2,2} & c_{2,2} \end{pmatrix}$$

is a representation of the lopsp operations. We already proved in Section 4 that this implies that the inflation matrix $R(o)$ is a representation too, i.e.

$$\begin{pmatrix} v_{o(G)} \\ e_{o(G)} \\ f_{o(G)} \end{pmatrix} = \begin{pmatrix} c_{0,0} & 4d_0 + c_{0,1} & c_{0,2} \\ 0 & 4d_0 + 4d_2 + 2 - c_{1,1} & 0 \\ 1 - c_{0,0} & 4d_2 + 1 - c_{0,1} - c_{0,2} & 1 - c_{0,2} \end{pmatrix} \begin{pmatrix} v_G \\ e_G \\ f_G \end{pmatrix}.$$

Therefore, Lemma 3.18 is true for lopsp operations too.

Lemma 5.17. *An lopsp operation preserves the Euler characteristic.*

6 Chiral Conway operations

There are only two chiral basic Conway operations: snub and gyro. There is one chiral extended Conway operator developed by George Hart [Har98], which is propeller. In Table 5.1 the double chamber patches and decorations for these operations are given.

For an lopsp operation o, the lopsp operation od has the same double chamber decoration with v_0 and v_2 swapped. This is not always easy to recognize in a double chamber patch. The lopsp operations sd is the *chiral pair* of snub. The lopsp operation pd is the dual of propeller, so $pd = dp$ and therefore $dpd = p$.

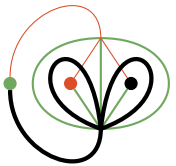
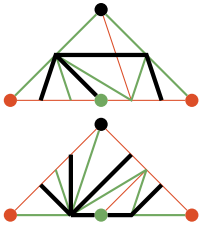
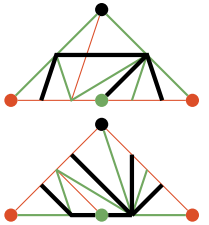
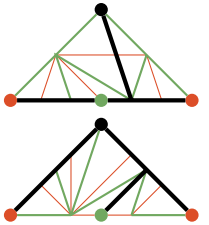
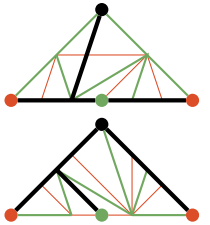
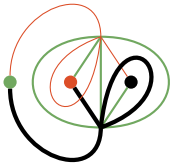
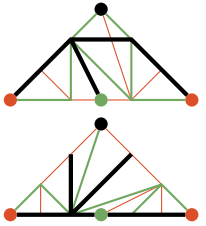
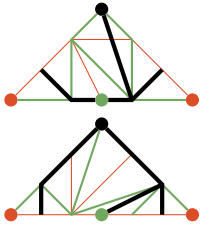
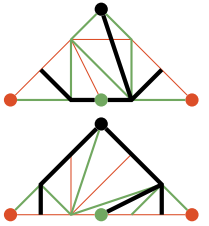
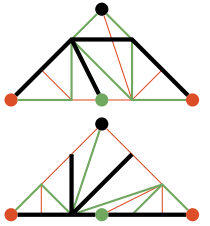
inflation factor	decoration	x	xd	dx	dxd
5					
		snub			gyro
5					
		propeller			

Table 5.1. The chiral Conway operations. There are two equivalent double chamber patches for each operation.

References

[Gol37] M. Goldberg. “A Class of Multi-Symmetric Polyhedra”. In: *Tohoku Mathematical Journal*. First Series 43 (1937), pp. 104–108.

[Har98] G.W. Hart. *Conway Notation for Polyhedra*. 1998. URL: http://www.georgehart.com/virtual-polyhedra/conway_notation.html.

[Men27] K. Menger. “Zur allgemeinen Kurventheorie”. In: *Fundamenta Mathematicae* 10.1 (1927), pp. 96–115.

6

Generation of local orientation-preserving symmetry preserving operations

In this chapter we develop an algorithm to generate all lopsp operations, similar to the algorithm for lsp operations in Chapter 4. We will again split the generation into two phases by first generating an underlying, coarser family of structures which we call *double predecorations*. These turn out to be more difficult to generate than predecorations as in this case there is no outer face to which we can add new vertices. On the other hand, we will see that this makes it easier to complete a double predecoration to a double chamber decoration.

1 Double predecorations

Since the degree of every type-1 vertex of a double chamber decoration D different from v_1 is 4, and the degree of all type-0 and type-2 vertices different from v_0, v_1 and v_2 is at least 6, the type-1 subgraph is almost a quadrangulation of the sphere with minimum degree 3. Only if v_1 is of type 1, the type-1 subgraph contains a face of size 2, which we can make into a quadrangle by adding an extra edge.

Lemma 6.1. *Let D be a double chamber decoration. If v_1 is not of type 1, let P be the type-1 subgraph of D. Otherwise, let P be the type-1 subgraph of D augmented with one edge from v_1 to an arbitrary neighbour in D. Then P is a quadrangulation of the sphere and there are at most three vertices of degree less than 3 and at most two vertices of degree 1 with a neighbour of degree 3.*

Proof. The degree of v_1 in D is 2 if and only if it has type 1. In that case, there is a face of size 2 in the type-1 subgraph since D is a triangulation. By adding one extra edge to the subgraph between v_1 and one of the vertices in the face of size 2, this face becomes a (degenerate) quadrangle (see Figure 6.1). Since all other type-1 vertices in D have degree 4, we find that P is a quadrangulation of the sphere.

Figure 6.1. A vertex v_1 of degree 2 in D, and the corresponding (degenerate) quadrangle in P with an extra edge.

Only if a vertex of type 1 or 2 in D has degree at most 4, this vertex has degree less than 3 in the type-2 subgraph. This is only possible in v_0, v_1 and v_2, so there are at most three vertices of degree less than 3 in P. If there are three vertices of degree 1 with a neighbour of degree 3 in the (modified) type-2 subgraph, they are v_0, v_1 and v_2. But then the neighbour of v_1 has only degree 4 in D, and therefore is v_0 or v_2, which is impossible. □

This result forms the motivation for the following definition.

Definition 6.2. A *double predecoration* is a quadrangulation of the sphere, possibly with double edges, with at most three vertices with degree less than 3 and at most two vertices of degree 1 with a neighbour of degree 3.

A problem with single chamber predecorations is that some of them can not be completed to chamber decorations. Double predecorations

do not have that problem, since Lemma 6.1 holds in the other direction too.

Lemma 6.3. *For each double predecoration P there exists at least one double chamber decoration D such that P is the type-1 subgraph of D, possibly with one extra edge if v_1 has type 1 (as in Lemma 6.1).*

Proof. Given a double predecoration P, we can choose three vertices v_0, v_1 and v_2. If there are vertices of degree 1 with a neighbour of degree 3, choose these vertices for v_0 and v_2 (there are at most two such vertices). If there are other vertices with degree less than 3, choose them for the remaining v_0, v_1 and v_2, or else choose arbitrary vertices. If v_1 has degree 1, it is contained in a quadrangle with two other vertices (one of them occurs two times in the quadrangle). Remove the incident edge and add two new edges between v_1 and the two other vertices.

Every other face in P is a quadrangle. Add one vertex of type 1 in each quadrangulation, with four new edges to the vertices of the quadrangle.

All edges of P get type 1, and the newly added edges get type 0 and 2 (alternating around each vertex). It is easily checked that this graph satisfies all conditions of Theorem 5.8. □

Just like for lsp operations, we want to be able to generate only the 2-connected or 3-connected lopsp operations.

Definition 6.4. A *2-connected* (resp. *3-connected*) *double predecoration* is the double predecoration of a 2-connected (resp. 3-connected) double chamber decoration.

2 Construction of double predecorations

We will construct the double predecorations using the canonical construction path method [McK98], starting from the base graphs in Figure 6.2, and using the extensions in Figure 6.3. The second base graph is not a double predecoration by itself, but is a necessary intermediate graph. Extensions 2c and 3 are similar to the extensions used

in [Bri+05] to construct all simple quadrangulations of the sphere, but we need different extensions to allow double edges and limit the number of vertices with degree less than 3.

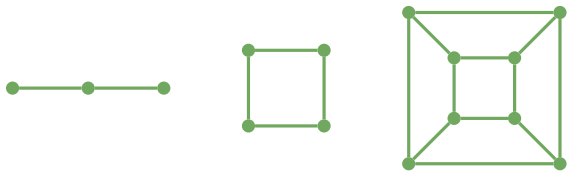


Figure 6.2. The base graphs.

We will now prove that each double predecoration can be constructed from the base graphs in Figure 6.2.

Lemma 6.5. *A quadrangulation of the sphere has minimum degree at most 3. If the minimum degree is 3, there are at at least 8 vertices of degree 3.*

Proof. Suppose G is a quadrangulation of the sphere with k vertices of degree 3 and no vertices of degree less than 3, and with V vertices, E edges and F faces. Since k vertices are incident to 3 edges, all other vertices are incident to at least 4 edges, and each edge is incident to exactly two vertices, we have $E \geq \frac{4V-k}{2}$. Each face in a quadrangulation contains four edges – possibly with some edges occuring two times in the same face. Since each edge has two sides and is therefore counted twice, we have $E = \frac{4F}{2} = 2F$. From Euler's formula (see Definition 2.8) we get

$$2 = V - E + F \leq \frac{2E + k}{4} - E + \frac{E}{2} = \frac{k}{4},$$

which implies $k \geq 8$. □

Lemma 6.6. *A 2-cycle in a quadrangulation of the sphere with minimum degree 3 contains at least 6 vertices of degree 3 on each side.*

Proof. For each of the two sides of the 2-cycle, we can remove all vertices on the other side, and one edge of the 2-cycle. The resulting subgraph is still a quadrangulation, and therefore contains at least 8

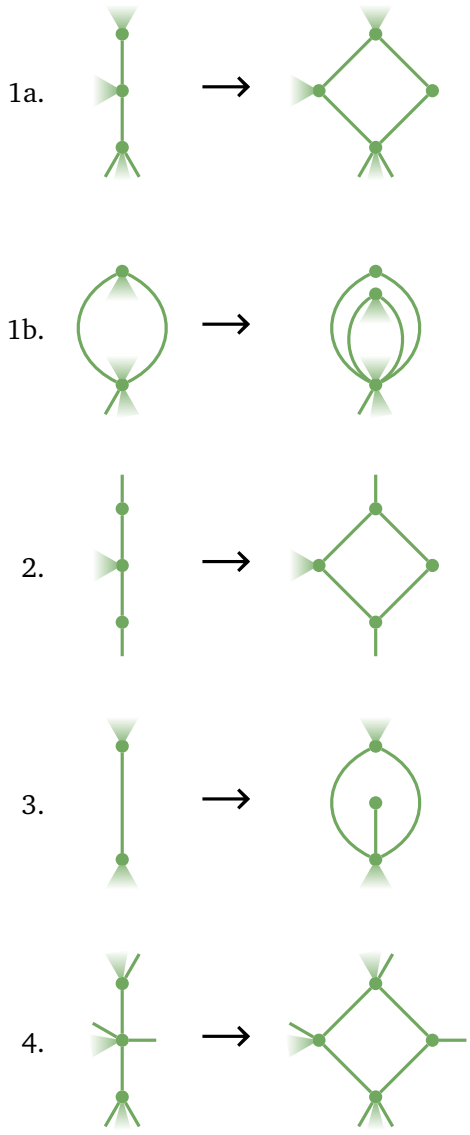


Figure 6.3. The extensions (from left to right) and reductions (from right to left). The shaded triangles contain zero or more edges.

vertices of degree 3 according to Lemma 6.5. Only 2 of these vertices can be vertices of the 2-cycle, so there are at least 6 other vertices of degree 3 on each side of the 2-cycle. □

Theorem 6.7. *Each double predecoration P different from the base graphs in Figure 6.2 can be reduced with one of the reductions in Figure 6.3 to a quadrangulation of the sphere with less vertices.*

Proof. The minimum degree of P is 1, 2 or 3 according to Lemma 6.5. If there is a vertex of degree 1, there are two possibilities (see Figure 6.4). Either P is a path of length 2 (i.e. a base graph), or P can be reduced by Reduction 3.

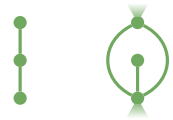


Figure 6.4. Double predecorations with minimum degree 1.

Suppose the minimum degree of P is 2. If P contains a path consisting of three consecutive vertices of degree 2, then P is a 4-cycle, which is the second base graph. If there are only two adjacent vertices of degree 2, then at least one of them is adjacent to a vertex of degree at least 4. So we are in the first situation shown in Figure 6.5. If there is a vertex of degree 2 adjacent to vertices of degree at least 3, we are in one of the other situations shown in Figure 6.5. The first situation can be reduced by Reduction 1a, the second and third one by Reduction 1a or 2 depending on the degrees of the adjacent vertices, and the last one by Reduction 1b.

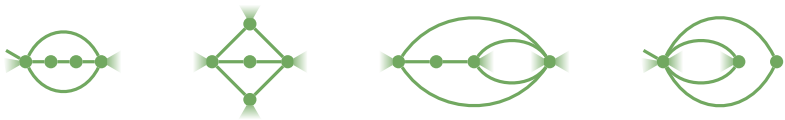


Figure 6.5. Double predecorations with minimum degree 2.

Finally, if the minimum degree is 3, there is at least one vertex v of degree 3. If two or more of the adjacent vertices of v coincide, we get the first or second situation of Figure 6.6. So there is at least

one 2-cycle, and this 2-cycle contains at least 6 vertices of degree 3 according to Lemma 6.6. We call one of these vertices v'. If the adjacent vertices of v' coincide, there is another 2-cycle completely contained in the previous one and therefore containing strictly fewer vertices. So we can prove by induction that there is always a vertex of degree 3 with three different adjacent vertices, as in the third situation of Figure 6.6.

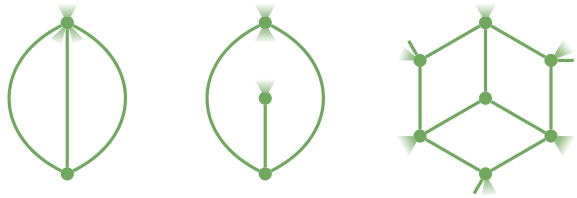


Figure 6.6. Double predecorations with minimum degree 3.

If there is a neighbour of v with degree more than 3, we can apply Reduction 4. Otherwise, if one of the three quadrangles containing v contains a vertex with degree more than 3, we can apply Reduction 4 on that quadrangle. If all the vertices in the figure have degree 3, then P is the third base graph of Figure 6.2. □

Given a double predecoration or intermediate graph G, we choose a *canonical parent* of G that is obtained by one of the reductions in Figure 6.3. We will always use the reduction with the smallest number – i.e. the first one in the list – among all possible reductions. It is possible that there is more than one way to apply this reduction to G, and if G has non-trivial symmetry, some of these can result in isomorphic parents.

The order of the extensions is chosen in such a way that the number of vertices with degree less than 3 can be kept low. However, some extensions can decrease this number, so it is possible that in order to construct a certain double predecoration, *intermediate graphs* with more than three vertices of degree less than 3 are required. We will prove in Lemma 6.9 that we need at most four such vertices. Therefore, our algorithm unfortunately does not construct double predecorations only, and some graphs with four vertices of degree less than 3 have to be filtered out at the end. Without Lemma 6.9, intermediate graphs

	vertices of degree		
extension	1	2	< 3
1a.	0	+1	+1
	−1	+2	+1
	0	0	0
1b.	0	+1	+1
2.	0	−1	−1
3.	0	+1	+1
	+1	0	+1
	0	0	0
	+1	−1	0
	+1	−2	−1
4.	0	0	0
	0	−1	−1

Table 6.1. All possible effects on the number of vertices with degree less than 3 for each extension.

with more vertices of degree less than 3 would be needed to construct some double predecorations, resulting in far more graphs that have to be filtered out later.

Lemma 6.8. *An extension applied to an intermediate graph G with k vertices of degree less than 3 results in a double predecoration G' with $k-1$, k or $k+1$ vertices of degree less than 3.*

Proof. Since each face that can be created by an extension is a quadrangle and the extensions do not change the Euler characteristic, G' is a quadrangulation.

We can check all possible combinations of degrees for the vertices affected by the extensions and verify that the difference in number of vertices with degree less than 3 is between −1 and 1. The results are given in Table 6.1. □

Note that it is possible to relax Extension 4 to Extension 4′ from Figure 6.7, but this extension sometimes decreases the number of

vertices with degree 2 by two. The result is that we have to allow intermediate graphs with up to five vertices of degree less than 3. So the algorithm has to construct more intermediate graphs, and some of them will not lead to a valid double predecoration. Such a behaviour is often detrimental for the generation rate (i.e. the number of generated structures per second) as much time is spent in structures that lead to dead ends in the generation process.

It is possible to combine Extensions 1a and 2 into 1′ (see Figure 6.7), but then consecutive applications of Extension 1′ can decrease the number of vertices of degree less than 3 by an arbitrary number. The order of Extensions 1a and 2 ensures that the second extension will not be canonical, as we will prove in Lemma 6.9.

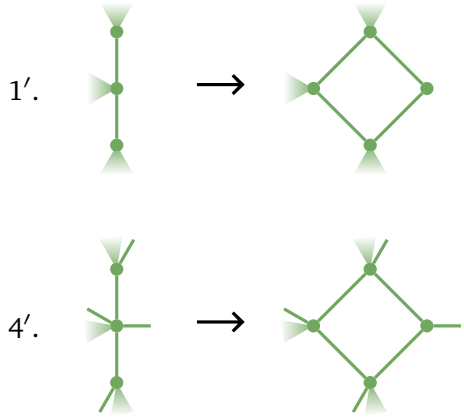


Figure 6.7. Alternative extensions.

Lemma 6.9. *A series of canonical reductions applied to a double predecoration can never result in an intermediate graph with more than four vertices of degree less than 3.*

Proof. Suppose there is a canonical reduction that increases the number of vertices with degree less than 3 to four. We know from Table 6.1 that this is only possible for Reductions 2, 3 and 4.

For Reduction 2 there are three possible situations, given in Figure 6.8. The first situation results in a base graph, so there is no subsequent

reduction that can increase the number of vertices with degree less than 3 to more than four. The second situation results in a graph with two vertices of degree 2, that can be reduced by Reduction 1a. Since this is the reduction with the smallest number it will be the next canonical reduction, and the number of vertices with degree less than 3 will decrease again. The third situation results in two vertices of degree 2 that can be reduced by Reduction 1b. So the next canonical reduction will be either 1a or 1b, both of which decrease the number of vertices with degree less than 3.

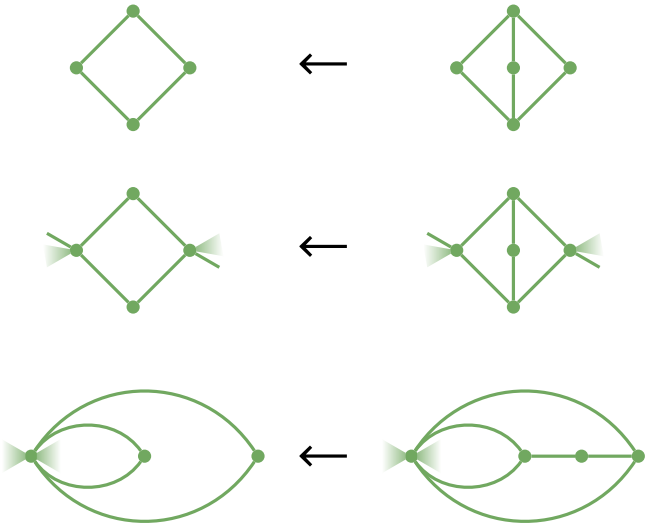


Figure 6.8. Situations where Reduction 2 increases the number of vertices with degree less than 3.

Reduction 3 can increase the number of vertices with degree less than 3 to four in the two situations given in Figure 6.9. The first situation results in a base graph, so there is no subsequent reduction. The second situation results in a graph with two vertices of degree 2 that can be reduced by Reduction 1a, so the next canonical reduction will decrease the number of vertices with degree less than 3.

The only other reduction that can increase the number of vertices with degree less than 3 is Reduction 4. But Reduction 4 is only canonical if there are no vertices of degree less than 3, otherwise one of the other

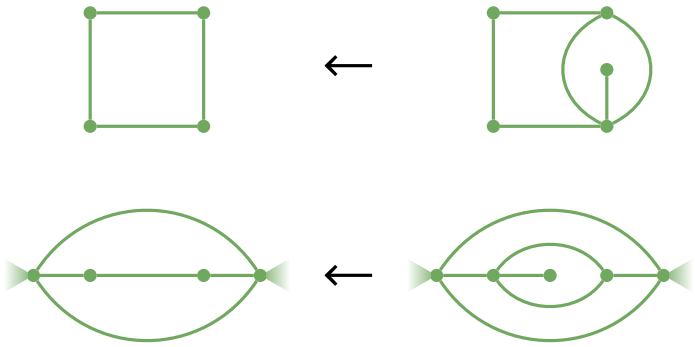


Figure 6.9. Situations where Reduction 3 increases the number of vertices with degree less than 3.

reductions can be applied. Therefore, Reduction 4 can only increase the number of vertices with degree less than 3 from zero to one, and not to four.

We proved that every time a canonical reduction increases the number of vertices with degree less than 3 to four, the result is a base graph or the next canonical reduction decreases the number back to three. Therefore, a series of canonical reductions never increases the number of vertices with degree less than 3 to more than four. □

3 Construction of 2-connected double predecorations

It follows immediately from Lemma 5.10 that a 2-connected double predecoration P is a double predecoration with no 2-cycles with three vertices of degree less than 3 on the same side. The two vertices of a 2-cycle form a 2-cut. We can split P into simple components $P_1, \dots, P_k$ by duplicating these vertices as in Figure 6.10. It is impossible that there are more than 3 edges between two vertices, since each 2-cycle has to separate v_0, v_1 and v_2.

We construct a bipartite auxiliary graph for P as follows. There are two types of vertices: the components $P_1, \dots, P_k$ form one type of vertex, and the pairs of vertices that are part of a 2-cycle (we call

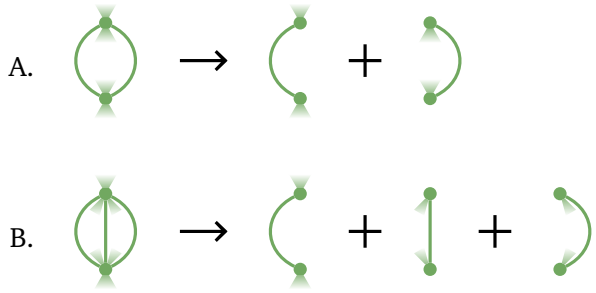


Figure 6.10. Splitting a double predecoration on a 2-cycle.

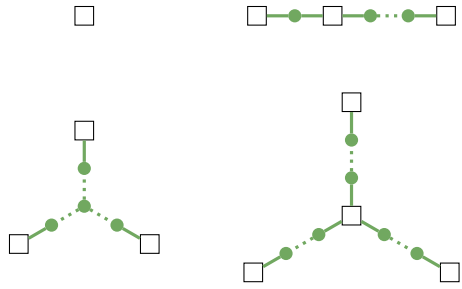


Figure 6.11. The four types of component trees. A component is represented by a square, and a cut by a point.

such a pair a *cut*) form the second type of vertex. Two vertices of the same type are never connected, and two vertices of a different type are connected if the component contains the vertices in the cut. Since each cut corresponds to a cut-vertex of the auxiliary graph, it is easily seen that this auxiliary graph is a tree. We call this the *component tree* of P. Since each 2-cycle separates v_0, v_1 and v_2, the component tree has at most 3 leaves, and each leaf contains at least one of v_0, v_1 and v_2. So the four types of component trees in Figure 6.11 are the only possibilities.

We can construct the 2-connected double predecorations by constructing all possible component trees. Each component is a simple quadrangulation of the sphere, and there can be at most three components with minimum degree lower than 3. Simple quadrangulations of the

sphere and simple quadrangulations of the sphere with minimum degree 3 can be generated by plantri [BM07; Bri+05]. Since plantri uses C_4 as base[1], the first base graph of Figure 6.2 should be added manually.

Each component is connected to other components by at most three cuts. If there is no cut, the component tree consists of a single component, which is easy to construct. If there is one cut, we can consecutively select one directed edge in each edge orbit as location for the cut. If there are two or three cuts, we can consecutively select one pair or triple of directed edges in each orbit of edge pairs or triples.

For a cut of type A (see Figure 6.10), we join the selected directed edges of the two adjacent components in the same direction. For a cut of type B (see Figure 6.10), which has cyclic symmetry, we can join the selected directed edges of the three adjacent components in two different orders.

We still have to take the symmetry of the component tree into account. If the component tree is a path graph, we obtain isomorphic double predecorations by taking the inverse edge of each selected edge. We can prevent this by choosing one component for which the selected edge or edges are not in the same orbit as their inverse edge or edges, and selecting only one of them. If there is no such component, we already construct only one double predecoration for this selection of edges. The double predecoration obtained by reversing the sequence of components in the path graph is isomorphic too. Since we can number the components increasingly in the order that plantri generates them, we can solve this by only using the lexicographically lowest sequence.

If the component tree has a cut of degree 3, we can obtain an isomorphic double predecoration by selecting the inverse edges and reversing the order of the components in the 3-cut. We solve this by using only one order in the 3-cut. Since rotating the component tree around the 3-cut results in an isomorphic double predecoration too, we again take the lexicographically lowest one.

[1] In [Bri+05] a quadrangulation of the sphere is defined as “a finite simple graph embedded on the sphere such that every face is bounded by a 4-cycle”.

The situation for a component with three cuts is similar, but now the symmetry depends on the symmetry of the three selected edges.

4 Construction of 3-connected double predecorations

It follows immediately from Lemmas 5.10 and 5.11 that a 2-connected double predecoration is a double predecoration with no 2-cycles with three vertices of degree less than 3 on the same side, and a 3-connected double predecoration is a double predecoration with no 2-cycles or separating 4-cycles with three vertices of degree less than 3 on the same side.

For 3-connected double predecorations, we take a similar approach. Each 2-cycle and separating 4-cycle of a 3-connected double predecoration P forms a cut. We can split P into simple components $P_1, \ldots, P_k$ by deduplicating these vertices as in Figures 6.10 and 6.12.

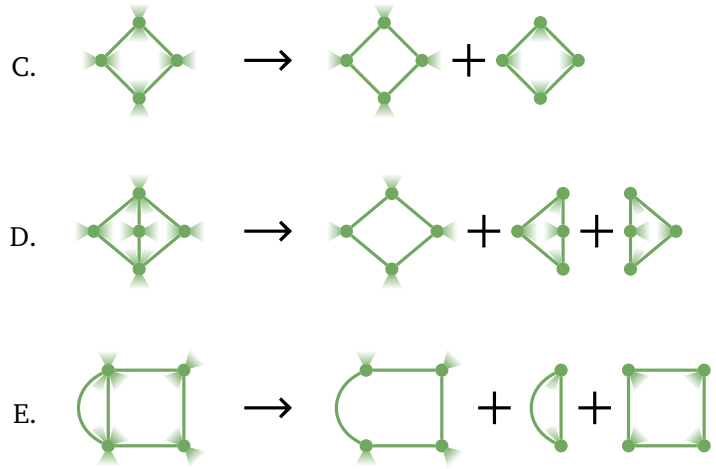


Figure 6.12. Splitting a double predecoration on 2-cycles and separating 4-cycles.

We represent each component $P_1, \ldots, P_k$ and each 2-cycle or separating 4-cycle by a vertex in the component tree. The component tree

has at most 3 leaves, and each leaf contains at least one of v_0, v_1 and v_2. So the four types of component trees in Figure 6.11 are still the only possibilities, but there are more types of cuts.

The components are simple quadrangulations of the sphere without separating 4-cycles, and only three of them can have minimum degree less than 3. These with minimum degree at least 3 can be generated by plantri.

Lemma 6.10. *The quadrangulations in Figure 6.13 are the only simple quadrangulations with no separating 4-cycles and minimum degree less than 3.*

Proof. According to Lemma 3 in [Bri+05], a simple quadrangulation with at least 6 vertices and no separating 4-cycles is 3-connected, and a 3-connected graph has minimum degree at least 3. It is easily checked that there are only two quadrangulations with less than 6 vertices and minimum degree less than 3. □

Figure 6.13. Quadrangulations with no separating 4-cycles.

In order to generate the 3-connected double predecorations isomorphism-free, we again have to take the symmetry of the component tree and cuts into account. Since there are more possibilities for the cut, this is a little bit more involved, but the principle is the same as for the 2-connected double predecorations.

5 Completion of double predecorations

As proven in Lemma 6.1, each double predecoration can be completed to at least one double chamber decoration. We can generate all double chamber decorations of a double predecoration with Algorithm 3. For a double chamber decoration with v_1 of type 0 or 2, the corresponding double predecoration is unique. If v_1 has type 1, there are possibly

two corresponding double predecorations, so we have to choose a canonical parent.

Algorithm 3 Complete a double predecoration in all possible ways.

1. Give all edges type 1.
2. Label all vertices of degree less than 3 with v_0, v_1 or v_2 in all non-isomorphic ways. For 3-connected double chamber decorations, make sure that v_1 is not separated from v_0 and v_2 by a 4-cycle.
3. If there are less than 3 vertices of degree less than 3, choose other vertices for the remaining v_0, v_1 and v_2 in all non-isomorphic ways.
4. If v_1 has degree 1, check whether it is the canonical parent, remove the incident edge and add two new edges.
5. Add a new type-1 vertex in each quadrangle and connect it to the vertices of the quadrangle using new edges.
6. Label the new edges with type 0 and 2 in the two possible ways.

We can now construct all double chamber decorations, but these are not the ideal structures if we want to apply the corresponding lopsp operations to an embedded graph. Double chamber patches are better suited for that. A double chamber decoration can have more than one corresponding double chamber patch, but since they are all equivalent, it suffices to construct one. We can cut a double chamber decoration D open along a simple path between v_1 and v_2 through v_0. According to Lemma 5.4, this always exists.

If we add a vertex t to the underlying graph[2] of D connected to v_1 and v_2, and call this graph G, finding such a simple path is equivalent to finding two vertex-disjoint paths in G between v_0 and t. This can be seen as a *maximum flow* problem in the directed graph G' where each vertex v of G is replaced by the two vertices v_{in} that has incoming edges from all neighbours of v, and v_{out} that has outgoing edges to all neighbours of v, with one edge from v_{in} to v_{out}. If each edge of G' has capacity one, a flow of 2 between $v_{0,\text{out}}$ and t_{in} corresponds to two vertex-disjoint paths, since each vertex v — in particular the edge

[2]We do not need the embedding to find a simple path.

between v_{in} and v_{out} — can be used only once. This problem can be solved by the Ford-Fulkerson algorithm [FF56]. If we want to find a shortest path, we can use Suurballe's algorithm [Suu74]. Algorithm 4 is a specialized algorithm based on Ford-Fulkerson.

Algorithm 4 Find a simple path in double chamber decoration D.

1. Find a path P_1 from v_0 to v_1 with BFS.
2. Remove the directed edges in P from D.
3. Find a path P_2 from v_0 to v_2 with BFS.
4. Remove edges that occur in both P_1 and P_2 (see Figure 6.14).

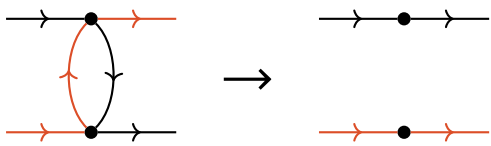


Figure 6.14. Removing an edge in P_1 and P_2.

6 Results

We used Algorithm 3 to construct the lopsp operations up to inflation factor 8. In Table 6.2, the chiral lopsp operations are given. This proves that there are no other chiral lopsp operations with inflation factor less than 7 than the chiral Conway operations in Table 5.1.

Each lopsp operation x has eight possible related lopsp operations (including itself): the compositions with dual dx, xd and dxd, its chiral pair x', and again the compositions with dual dx', $x'd$ and $dx'd$. Only for the second chiral operation with inflation factor 5 in Table 6.2, the operations dx and xd are equal. For the first chiral operations with inflation rates 5 and 7, the chiral pair x' is equal to xd. So these operations have only four related lopsp operations.

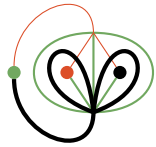
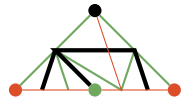
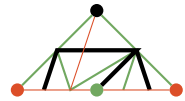
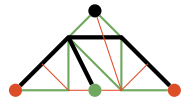
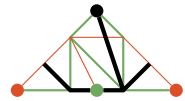
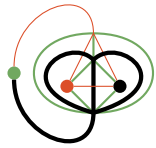
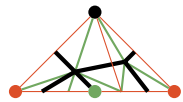
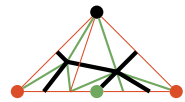
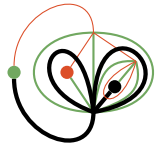
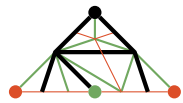
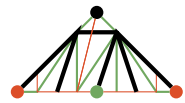
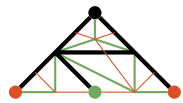
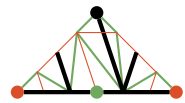
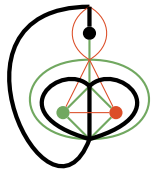
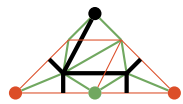
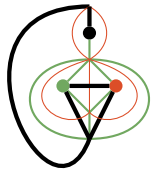
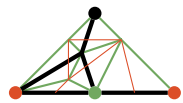
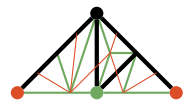
inflation factor	decoration	x	xd	n
5				4
				4
7				4
				8
				8
8				8
				8

Table 6.2. The chiral double chamber decorations with inflation factor up to 8, and the number n of related lopsp operations.

References

[BM07] G. Brinkmann and B.D. McKay. “Fast generation of planar graphs”. In: *MATCH Communications in Mathematical and in Computer Chemistry* 58.2 (2007), pp. 323–357.

[Bri+05] G. Brinkmann et al. “Generation of simple quadrangulations of the sphere”. In: *Discrete Mathematics* 305 (2005), pp. 33–54.

[FF56] L.R. Ford and D.R. Fulkerson. “Maximal Flow Through a Network”. In: *Canadian Journal of Mathematics* 8 (1956), pp. 399–404.

[McK98] B.D. McKay. “Isomorph-Free Exhaustive Generation”. In: *Journal of Algorithms* 26.2 (1998), pp. 306–324.

[Suu74] J.W. Suurballe. “Disjoint Paths in a Network”. In: *Networks* 4.2 (1974), pp. 125–145.

7

Open problems

In the previous chapters, we established a mathematical framework for local symmetry preserving operations and developed generation algorithms. This opens up possibilities for future research. In this chapter, we briefly sketch some open problems.

1 Symmetry generating polyhedra

Although symmetry preserving operations can be applied to any embedded graph, our primary motivation was to apply them to polyhedra with certain symmetries. We know the symmetry group of a polyhedron is a finite *point group* in three dimensions [Cox91].

Definition 7.1. A *point group* is a group of symmetries with at least one common fixpoint.

There are seven infinite families of finite point groups in three dimensions, called the *axial groups*, and seven other finite point groups, the *polyhedral groups*. See Table 7.1 for a description of each group.

Each polyhedral group is named after a polyhedron with that symmetry group. We already know most of these polyhedra from Chapter 1, except for the *pyritohedron* which has symmetry group T_h. The pyritohedron has the same underlying graph as the dodecahedron, but with less symmetry since not all edges have the same length (see Figure 7.1).

	Cyclic groups
C_n	one vertical n-fold rotation axis
$C_{n\mathrm{h}}$	C_n with a horizontal mirror plane
$C_{n\mathrm{v}}$	C_n with n vertical mirror planes
	"Spiegel" group
S_{2n}	one vertical $2n$-fold rotoreflection axis
	Dihedral groups
D_n	C_n with n horizontal 2-fold rotation axes
$D_{n\mathrm{h}}$	D_n with a horizontal mirror plane
$D_{n\mathrm{d}}$	D_n with n vertical mirror planes between the rotation axes
	Polyhedral groups
T	chiral tetrahedral symmetry
T_d	full tetrahedral symmetry
T_h	pyritohedral symmetry
O	chiral octahedral symmetry
O_h	full octahedral symmetry
I	chiral icosahedral symmetry
I_h	full icosahedral symmetry

Table 7.1. The finite point groups in three dimensions.

For some polyhedra P with symmetry group G, each chamber of the barycentric subdivision is a *fundamental domain*, i.e. contains exactly one point from each orbit. Therefore, each polyhedron Q with the same symmetry group can be constructed by applying an lsp operation to P, by taking the fundamental domain as chamber decoration. We say that P is a *generating polyhedron* for G.

Theorem 7.2. *The tetrahedron is a generating polyhedron for T_d, the cube and octahedron are generating polyhedra for O_h, and the dodecahedron and icosahedron are generating polyhedra for I_h.*

For some other polyhedra, each double chamber is a fundamental domain. In that case, the symmetry group G is orientation preserving,

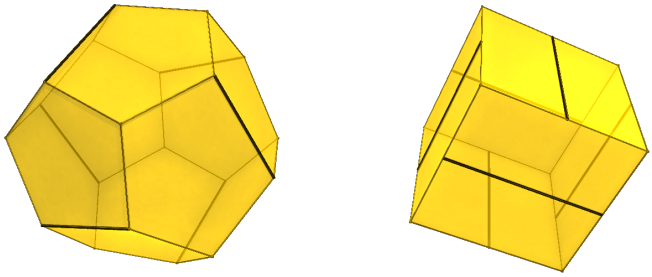


Figure 7.1. The pyritohedron is a polyhedron that resembles the dodecahedron, but there are 6 congruent edges with a different length. A degenerate case is the cube with one extra edge dividing each face. This is technically not a convex polyhedron since there are two faces in the same plane.

and for each polyhedron Q with the same symmetry group there is an lopsp operation o such that $Q = oP$.

Theorem 7.3. *The tetrahedron is a generating polyhedron for T, the cube and octahedron are generating polyhedra for O, and the dodecahedron and icosahedron are generating polyhedra for I.*

The only polyhedral group for which there is no generating polyhedron is T_h. A chamber or double chamber of the barycentric subdivision of the pyritohedron is not a fundamental domain.

For the axial groups, there are no generating polyhedra, but $D_{n\mathrm{h}}$ is a special case. If we allow polyhedra with faces of size two or vertices of degree two, the dual embedded graphs in Figure 7.2 are generating polyhedra for $D_{n\mathrm{h}}$. Note that some lsp operations applied to these embedded graphs result in embedded graphs with faces of size two or vertices of degree two, so if we want to generate all polyhedra with $D_{n\mathrm{h}}$ symmetry, we have to ignore these.

Thus, using the algorithms of Chapters 4 and 6 we can easily generate all polyhedra with symmetry groups T_{d}, O_{h}, I_{h}, T, O, I and $D_{n\mathrm{h}}$ from one base graph. For other symmetry groups, e.g. the trivial group, we obviously need infinitely many base graphs.

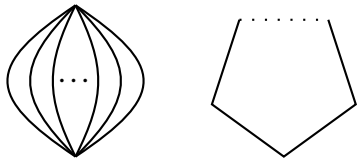


Figure 7.2. Degenerate polyhedra with dihedral symmetry.

Question 1. *Do there exist other symmetry groups for which we can generate all polyhedra from a finite number of base graphs?*

2 Symmetry increasing operations

As already mentioned in Remark 1.5, the lsp operation ambo can increase the symmetry of a polyhedron [OPW10; Hub+13]. If we apply ambo to the tetrahedron, the result is an octahedron. Of course, operations that can be written as ambo followed by another operation can increase the symmetry too. Snub is an lopsp operation that can increase the symmetry. Applied to the tetrahedron, it results in an icosahedron. Consequently, the dual operation gyro results in a dodecahedron.

Question 2. *Can a sequence of operations containing neither ambo nor snub increase the symmetry of polyhedra?*

For every lsp operation o and polyhedron P with symmetry group G, the symmetry group of oP is a supergroup of G. Since G is a finite point group in three dimensions, we know all possible supergroups. In Figure 7.3 the subgroup hierarchy of the polyhedral groups is given.

For non-polyhedral embedded graphs, there are more operations that can increase the symmetry. If we apply the lsp operation in Figure 7.4 to a 4-regular toroidal quadrangulation with symmetry group $\mathbb{Z}_m \times \mathbb{Z}_n$, the resulting symmetry group is $\mathbb{Z}_{3m} \times \mathbb{Z}_{3n}$. Another example is chamfer, which increases the symmetry of embedded graphs with only hexagonal faces and vertices of degree 3 [Río14]. These cannot be embedded in the plane, and are therefore not polyhedral, but they can be embedded on the torus.

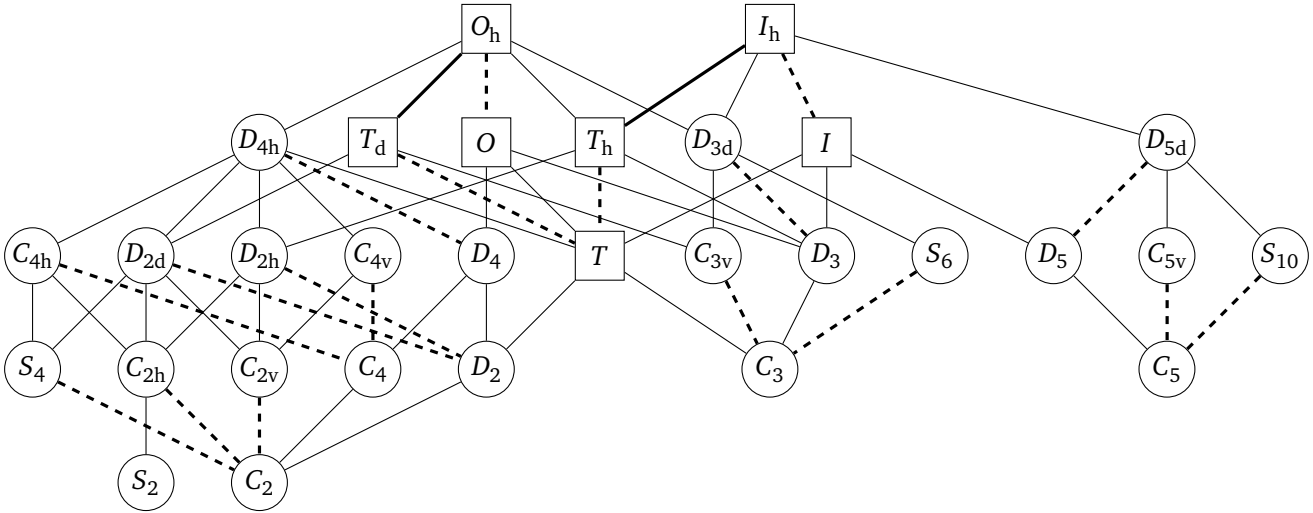


Figure 7.3. Subgroup hierarchy for the polyhedral groups. The squares are the polyhedral groups, and the circles are axial groups. The dashed lines are symmetries that can be decreased by chiral lopsp operations. The thick solid lines are symmetries that can be increased by ambo and snub.

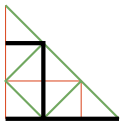


Figure 7.4. An lsp operation that can increase the symmetry of toroidal embedded graphs.

Question 3. *What are the symmetry increasing operations for embedded graphs with higher genus?*

3 Decomposition of operations

In Chapters 3 and 5 the composition operation for lsp and lopsp operations was introduced. We observed that some operations are simple, and others are composite. For most lsp and lopsp operations, there seems to be a unique decomposition in simple operations if we ignore intermittent dual operations. But some operations commute with ambo, and Figure 7.5 gives an infinite class of lsp operations that is a commutative submonoid of the lsp operations.

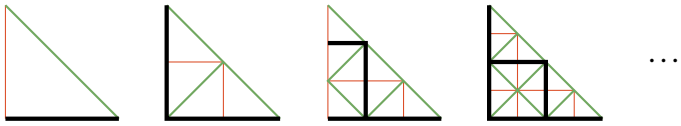


Figure 7.5. Mutually commutative lsp operations.

Question 4. *Which operations can commute with others? Can we still prove some kind of uniqueness for decomposition? What is the connection between commuting and symmetry increasing lsp operations?*

Another problem with decomposition is that it is not easy to decompose a given operation. The most obvious algorithms are exponential in the size of the operation.

Question 5. *Does there exist a polynomial algorithm for decomposition?*

Related to the decomposition of operations, is the decomposition of an embedded graph G in an operation o and a smaller embedded graph

G' such that $G = oG'$. Similar to how a symmetric embedded graph can be completely described by its fundamental domain and symmetry group, thereby reducing the amount of redundant information, we can describe an embedded graph by giving a smaller embedded graph and an operation. The smaller graph plays the role of the symmetry group, and the operation the role of the fundamental domain. Even for some embedded graphs with trivial symmetry, for which the fundamental domain is the whole graph, this decomposition can find the hidden structure.

4 Connectivity of graphs with higher genus

We investigated the effect of lsp and lopsp operations on the connectivity of plane graphs in Theorems 3.4 and 5.9. As mentioned in Remark 3.5, the effect on embedded graphs with higher genus can be completely different.

Question 6. *Which operations can decrease the connectivity?*

For 3-connected lsp operations, this question is answered in [Cam20].

References

[Cam20] H. Van den Camp. "The Effect of Local Symmetry Preserving Operations on the Connectivity of Embedded Graphs". Master's Thesis. Ghent University, 2020.

[Cox91] H.S.M. Coxeter. *Regular Complex Polytopes*. 2nd ed. Cambridge University Press, 1991.

[Hub+13] I. Hubard et al. "Medial Symmetry Type Graphs". In: *The Electronic Journal of Combinatorics* 20.3 (2013), p. 29.

[OPW10] A. Orbanić, D. Pellicer and A.I. Weiss. "Map operations and k-orbit maps". In: *Journal of Combinatorial Theory, Series A* 117.4 (2010), pp. 411–429.

[Río14] M. del Río Francos. "Chamfering operation on k-orbit maps". In: *Ars Mathematica Contemporanea* 7.2 (2014), pp. 519–536.

Bibliography

[BBZ18] D. Bokal, G. Brinkmann and C.T. Zamfirescu. *The Connectivity of the Dual*. 2018. arXiv: 1812.08510.

[BGS17] G. Brinkmann, P. Goetschalckx and S. Schein. "Comparing the constructions of Goldberg, Fuller, Caspar, Klug and Coxeter, and a general approach to local symmetry-preserving operations". In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 473.2206 (2017).

[BM07] G. Brinkmann and B.D. McKay. "Fast generation of planar graphs". In: *MATCH Communications in Mathematical and in Computer Chemistry* 58.2 (2007), pp. 323–357.

[Bri+05] G. Brinkmann et al. "Generation of simple quadrangulations of the sphere". In: *Discrete Mathematics* 305 (2005), pp. 33–54.

[Cam20] H. Van den Camp. "The Effect of Local Symmetry Preserving Operations on the Connectivity of Embedded Graphs". Master's Thesis. Ghent University, 2020.

[Cas84] D.L.D. Caspar. "This Week's Citation Classic". In: *Current Contents Life Sciences* 4 (1984), p. 168.

[CK62] D.L.D. Caspar and A. Klug. "Physical Principles in the Construction of Regular Viruses". In: *Cold Spring Harbor Symposia on Quantitative Biology*. Vol. 27. 1962, pp. 1–24.

[CK63] D.L.D. Caspar and A. Klug. “Structure and Assembly of Regular Virus Particles”. In: *Viruses, Nucleic Acids and Cancer. A Collection of Papers Presented at the 17th Annual Symposium on Fundamental Cancer Research*. The Williams and Wilkins Company, 1963, pp. 27–39.

[Cat65] M.E. Catalan. “Mémoire sur la théorie des polyèdres”. In: *Journal de l’école Impériale Polytechnique* XLI (1865), pp. 1–71.

[CBG08] J.H. Conway, H. Burgiel and C. Goodman-Strauss. *The Symmetries of Things*. A K Peters, 2008.

[Cox71] H.S.M. Coxeter. “Virus Macromolecules and Geodesic Domes”. In: *A Spectrum of Mathematics. essays presented to H.G. Forder*. Ed. by J.C. Butcher. Auckland University Press, 1971, pp. 98–107.

[Cox73] H.S.M. Coxeter. *Regular Polytopes*. 3rd ed. Dover Publications, 1973.

[Cox91] H.S.M. Coxeter. *Regular Complex Polytopes*. 2nd ed. Cambridge University Press, 1991.

[CW56] F.H.C. Crick and J.D. Watson. “Structure of Small Viruses”. In: *Nature* 177 (1956), pp. 473–475.

[Die97] R. Diestel. *Graph Theory*. Graduate Texts in Mathematics 173. Springer-Verlag, 1997.

[DH87] A.W.M. Dress and D. Huson. “On tilings of the plane”. In: *Geometriae Dedicata* 24.3 (1987), pp. 295–310.

[DD04] M. Dutour and M. Deza. “Goldberg–Coxeter Construction for 3- and 4-valent Plane Graphs”. In: *The Electronic Journal of Combinatorics* 11 (2004).

[Eec33] P. Ver Eecke. *Pappus d’Alexandrie: La Collection Mathématique*. Desclée de Brouwer, 1933.

[FF56] L.R. Ford and D.R. Fulkerson. “Maximal Flow Through a Network”. In: *Canadian Journal of Mathematics* 8 (1956), pp. 399–404.

[FM06] P.W. Fowler and D.E. Manolopoulos. *An Atlas of Fullerenes*. 2nd ed. Dover Publications, 2006.

[FP94] P.W. Fowler and T. Pisanski. “Leapfrog Transformations and Polyhedra of Clar Type”. In: vol. 90. 19. 1994, pp. 2865–2871.

[GR01] C. Godsil and G. Royle. *Algebraic Graph Theory*. Graduate Texts in Mathematics 207. Springer, 2001.

[Goe19] P. Goetschalckx. *decogen*. 2019. URL: https://github.com/314eter/decogen.

[GCC20a] P. Goetschalckx, K. Coolsaet and N. Van Cleemput. “Generation of Local Symmetry-Preserving Operations”. In: *Ars Mathematica Contemporanea* (2020). eprint: 1908.11622.

[GCC20b] P. Goetschalckx, K. Coolsaet and N. Van Cleemput. “Local Orientation-Preserving Symmetry Preserving Operations on Polyhedra”. In: *Discrete Mathematics* (2020). eprint: 2004.05501.

[Gol37] M. Goldberg. “A Class of Multi-Symmetric Polyhedra”. In: *Tohoku Mathematical Journal*. First Series 43 (1937), pp. 104–108.

[GT01] J.L. Gross and T.W. Tucker. *Topological Graph Theory*. Dover Publications, 2001.

[GYZ14] J.L. Gross, J. Yellen and P. Zhang, eds. *Handbook of Graph Theory*. 2nd ed. CRC Press, 2014.

[Grü94] B. Grünbaum. “Polyhedra with hollow faces”. In: *Proceedings of the NATO Advanced Study Institute on Polytopes: Abstract, Convex and Computational*. Ed. by T. Bisztriczky et al. Springer, 1994, pp. 43–70.

[Grü03] B. Grünbaum. *Convex Polytopes*. Ed. by V. Kaibel, V. Klee and G.M. Ziegler. 2nd ed. Graduate Texts in Mathematics 221. Springer, 2003.

[GS87] B. Grünbaum and G.C. Shephard. *Tilings and Patterns*. W. H. Freeman and Company, 1987.

[GLM97] T. Grüner, R. Laue and M. Meringer. “Algorithms for group actions: homomorphism principle and orderly generation applied to graphs”. In: *Groups and Computation II*. Ed. by L. Finkelstein and W. Kantor. DIMACS Series in Discrete Mathematics and Theoretical Computer Science 28. 1997, pp. 113–122.

[Har98] G.W. Hart. *Conway Notation for Polyhedra*. 1998. URL: http://www.georgehart.com/virtual-polyhedra/conway_notation.html.

[Hub+13] I. Hubard et al. “Medial Symmetry Type Graphs”. In: *The Electronic Journal of Combinatorics* 20.3 (2013), p. 29.

[Joh66] N.W. Johnson. “Convex Polyhedra with Regular Faces”. In: *Canadian Journal of Mathematics* 18 (1966), pp. 169–200.

[Kep19] J. Kepler. *Ioannis Keppleri Harmonices mundi libri V*. Linz, 1619.

[Kro+85] H.W. Kroto et al. “C_{60}: Buckminsterfullerene”. In: *Nature* 318 (1985), pp. 162–163.

[Man71] P. Mani. “Automorphismen von polyedrischen Graphen”. In: *Mathematische Annalen* 192 (1971), pp. 279–303.

[MF73] R.W. Marks and R. Buckminster Fuller. *The Dymaxion world of Buckminster Fuller*. Doubleday Anchor Books, 1973.

[McK98] B.D. McKay. “Isomorph-Free Exhaustive Generation”. In: *Journal of Algorithms* 26.2 (1998), pp. 306–324.

[Men27] K. Menger. “Zur allgemeinen Kurventheorie”. In: *Fundamenta Mathematicae* 10.1 (1927), pp. 96–115.

[OPW10] A. Orbanić, D. Pellicer and A.I. Weiss. “Map operations and k-orbit maps”. In: *Journal of Combinatorial Theory, Series A* 117.4 (2010), pp. 411–429.

[PWB17] T. Pisanski, G. Williams and L.W. Berman. “Operations on Oriented Maps”. In: *Symmetry* 9 (2017), p. 274.

[Rec17] B.R.S. Recht. *Notes on operations on polyhedra*. 2017. URL: https://antitile.readthedocs.io/en/latest/conway.html.

[Río14] M. del Río Francos. "Chamfering operation on k-orbit maps". In: *Ars Mathematica Contemporanea* 7.2 (2014), pp. 519–536.

[Ste16] E. Steinitz. "Polyeder und Raumeinteilungen". In: *Encyclopädie der mathematischen Wissenschaften*. Ed. by W.F. Meyer and H. Mohrmann. Vol. 3.AB12. B.G. Teubner, 1916.

[Suu74] J.W. Suurballe. "Disjoint Paths in a Network". In: *Networks* 4.2 (1974), pp. 125–145.

[Whi32] H. Whitney. "Congruent Graphs and the Connectivity of Graphs". In: *American Journal of Mathematics* 54.1 (1932), pp. 150–168.

[Wyt18] W.A. Wythoff. "A relation between the polytopes of the C600-family". In: *Proceedings of the Section of Sciences* 20 (1918), pp. 966–970.

[Zal67] V.A. Zalgaller. "Convex Polyhedra with Regular Faces". In: *Zap. Nauchn. Semin. Leningr. Otd. Mat. Inst. Steklova* 2 (1967), pp. 1–221.

Index

Colophon

This book was typeset using the LaTeX typesetting system created by Leslie Lamport and the memoir class created by Peter Wilson. The body font is Bitstream Charter designed by Matthew Carter, which includes italics and small caps.