Discrete geometry and optimization

vorgelegt von
M. Sc.
Manuel Radons

an der Fakultät II – Mathematik und Naturwissenschaften
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften
Dr.rer.nat.

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr. Volker Mehrmann
Gutachter:	Prof. Dr. Michael Joswig
Gutachter:	Prof. Dr. Joseph O'Rourke
Gutachter:	Prof. Dr. Stéphane Gaubert

Tag der wissenschaftlichen Aussprache: 26.10.2022

Berlin 2022

Summary

This thesis analyses three problems from discrete geometry and optimization via discrete-analytic hybrid methods.

The absolute value equation (AVE) is a piecewise linear system which is equivalent to the linear complementarity problem (LCP) and thus generalizes linear and quadratic programming. Unique solvability for arbitrary right-hand sides has long been comprehensively characterized for both systems, but not non-unique solvability. To fill this gap in the theory, we analyze homotopies of the identity map and the piecewise linear function corresponding to the AVE. Doing so, we identify certain real eigenvalues of the coefficient matrix of the AVE that determine the singularity-structure of these homotopies. Let k be the number of these eigenvalues which is larger than one. We prove that the mapping degree of the aforementioned piecewise linear function is congruent to $(k+1) \bmod 2$. We also derive an exact formula for the degree, which is more technical. Finally, we develop the analogous results for the LCP.

The calibration of internal combustion engines is a problem that has risen to prominence during the so-called diesel crisis. Engine behavior can be modeled as a function from the space of actuators to the space of performance measurands. A calibration solution is a set of lookup tables that associate performance demands to points in the actuator space which achieve these demands, while respecting side constraints such as emission standards. We develop an adaptive multigrid method for identifying and circumventing loci of turbulence, which are the cause of emission spikes during the generation of the lookup tables. With the resulting algorithm we calibrate a benchmark engine model in accordance with several EURO emission norms.

Dürer's problem asks whether every 3-polytope has a net, that is, whether it is possible to cut it along some spanning tree of its edge graph so that the resulting connected surface may be unfolded flat into the plane without self-overlaps. A 3-prismatoid is the convex hull of two polygons A, B in parallel planes H_A and H_B. It is called nested if the orthogonal projection of A to H_B is properly contained in B, or vice versa. We show that every nested prismatoid has a net. To this end we adapt and extend an unfolding technique pioneered by O'Rourke in his work on nearly flat, acutely triangulated convex caps. The basic approach is to project a nested prismatoid P—a convex cap by definition—to the plane H_B, establish a cutting scheme for the flat polytope, lift the scheme back into P, and then prove that a certain continuous deformation of the unfolding of the flat polytope into the unfolding of P does not cause overlaps.

Zusammenfassung

Diese Arbeit behandelt drei Probleme aus der diskreten Geometrie und Optimierung mittels diskret-analytischer Hybridmethoden.

Die Absolutwertgleichung (AWG) ist ein stückweise lineares Gleichungssystem, das äquivalent zum linearen Komplementaritätsproblem (LCP) ist und daher lineare und quadratische Programmierung generalisiert. Die vollständige Charakterisierung eindeutiger Lösbarkeit von AWG und LCP für beliebige rechte Seiten sind klassische Resultate der Literatur. Eine vergleichbare Charakterisierung der nicht-eindeutigen Lösbarkeit existiert jedoch nicht. Um diese Lücke in der Theorie zu schließen, analysiseren wir Homotopien der Identitätsabbildung und der stückweise linearen Funktion, welche durch die AWG definiert wird. Hierbei identifizieren wir bestimmte reelle Eigenwerte der Koeffizientenmatrix der AWG, welche die Singulariätsstruktur der untersuchten Homotopien determinieren. Es sei k die Zahl der besagten Eigenwerte, welche größer als eins sind. Wir beweisen, dass in diesem Fall der Abbildungsgrad der stückweise linearen Funktion, welche der AWG korrespondiert, kongruent zu $(k+1) \bmod 2$ ist. Desweiteren leiten wir eine exakte Formel für den Abbildungsgrad her. Schließlich entwickeln wir die analogen Ergebnisse für das LCP.

Die Kalibrierung von Verbrennungsmotoren ist ein Problem, das während der sogenannten Dieselkrise zur Prominenz gelangte. Das Motorenverhalten kann als eine Funktion vom Raum der Aktuatoren in den Raum der Leistungsmesswerte modelliert werden. Eine Kalibrierungslösung ist ein Satz von Lookup-Tabellen, die einer Menge von Leistungsanforderungen eine Menge von Punkten im Aktuatorenraum zuordnet, welche besagte Anforderungen, unter Berücksichtigung von Nebenbedingungnen wie z.B. Emissionsgrenzen, realisieren. Wir entwickeln ein adaptives Mehrgitterverfahren um bei der Erstellung der Kalibrierungslösung Regionen stark nichtlinearen Motorenverhaltens im Aktuatorenraum, welche eine Hauptursache von Emissionsspitzen sind, zu identifizieren und zu umgehen. Mittels des resultierenden Algorithmus kalibrieren wir ein Benchmark-Motorenmodell unter Einhaltung verschiedener EURO-Abgasnormen.

Dürers Problem stellt die Frage, ob jedes 3-Polytop ein Netz hat, das heißt, ob man es entlang eines Spannbaums seines Kantengraphen aufschneiden kann, so dass die resultierende zusammenhängende Fläche ohne Selbstüberlappungen in die Ebene aufgefaltet werden kann. Ein 3-Prismatoid ist die konvexe Hülle zweier Polygone in parallelen Ebenen H_A und H_B. Es wird verschachtelt genannt, falls die orthogonale Projektion von A auf H_B echt in B enthalten ist. Wir beweisen, dass jedes verschachtelte Prismatoid ein Netz hat. Hierzu adaptieren und erweitern wir eine Auffaltungsstrategie, die von O'Rourke eingeführt wurde. Der Grundgedanke ist, ein verschachteltes Prismatoid P nach H_B zu projizieren, ein Schnittschema für das flache Polytop einzuführen, nach P zurück zu projizieren und dann zu beweisen, dass eine bestimmte stetige Verformung der Auffaltung des flachen Polytops in die Auffaltung von P nicht zu Überlappungen führt.

Acknowledgements

I wish to thank my advisor Michael Joswig for his scientific guidance and the occasional motivational kick in the back. Further, I would like to thank my BMS-mentor Martin Skutella, the co-referees Joseph O'Rourke and Stéphane Gaubert, as well as my co-authors Timo Burggraf, Marc E. Pfetsch, Stefan Ulbrich, and Josue Tonelli-Cueto for productive discussions. A special thanks goes out to Günter M. Ziegler, who has, on several occasions, proved to be a selfless friend and mentor.

I owe a debt of gratitude to my colleagues, who made my time at TU a bliss. In particular, I wish to thank Marta Panizzut, Holger Eble, Oğuzhan Yürük, and Andrew Newman for laughs, support, and interesting conversations.

I wish to sincerely thank my friends and family, mom, dad, Tina, Thomas, Katharina, Elise, Christian, Lars and Jennifer for making me feel at home.

Finally, I want to express my great sadness that Andreas Griewank cannot celebrate this moment with me. His sudden death is a great loss to everyone who knew him.

Contents

Introduction

The fields of nonlinear optimization and discrete geometry naturally interact and enrich each other. For example, a frequent object of study in nonlinear optimization are functions which are *nonsmooth* in the sense that they are continuous and differentiable everywhere except on some lower dimensional surface or other “small” subsets of the domain. The cuts in the flow of derivative information caused by nonsmoothness turn analytical problems into hybrids of interwoven analytic and combinatorial issues that necessitate the incorporation of discrete methods into the toolbox for their investigation.

This is exemplified by generalized derivative concepts such as the Bouligand derivative, wherein the local linear approximation of a function—the classical derivative—is replaced by a *piecewise linear approximation* whose linear pieces are the directional derivatives, if they exist [Sch12, p. 67ff]. The combinatorial difficulty of the corresponding *piecewise linear system* – whose solution may be a step, e.g., in a generalized Newton’s method incorporating the Bouligand derivative – can often be bounded in terms of analytic properties of the underlying nonsmooth function. Coherent orientation, convexity, and local bijectivity are inherited by definition [Sch12, p. 67ff]. Other properties, such as local Lipschitz constants, have a more subtle influence on the structure of local piecewise linearizations; see, for example [Gri13], [GBRS15], [GSL$^+$18].

The key difficulty of solving a piecewise linear system is to determine the cell(s) of the corresponding *polyhedral domain-subdivision* containing the solution(s), of which there can be exponentially many. For functions with finitely many linear pieces this task can be reduced to solving a *linear complementarity problem* (LCP), or an equivalent *absolute value equation* (AVE) [GBRS15, Lem. 6.5.]. The *piecewise linear functions* corresponding to AVE and LCP are linear on the orthants of $\mathbb{R}^n$ and thus positively homogeneous. This allows to investigate them as spherical maps, e.g., via

$$\mathbb{S}^{n-1} \ni \ x \ \mapsto \ \frac{F(x)}{\|F(x)\|_2}\,,$$

making a straightforward application of topological tools such as *homotopy* and *degree theory* possible. Moreover, since the orthants of $\mathbb{R}^n$ are simplicial cones, key structural questions about the functions can be answered by means of basic linear algebra. Switching back and forth between the spherical and the simplicial view facilitates the proof of our first main result [RTC19].

Main Result 1. *Exact formulae and formulae* $\operatorname{mod} 2$ *for the degree of the piecewise linear functions associated to an LCP, resp., AVE.*

Piecewise linearizations can also occur in a context where the investigated function is not given by an analytic expression, but instead in the form of experimental samples.

In the second chapter of this thesis we develop an algorithm for the practical calibration of internal combustion engines. A *calibration solution* is a set of support points of a piecewise linear function, called an *engine map*, that outputs the necessary engine settings for a given performance demand. As indicated above, the engine behavior is not given as a finite formula or program, but can only be sampled. It has to be assumed that the relations between the actuator settings of an engine and its performance data is nonsmooth or at least not sufficiently smooth, since *smooth models* consistently fail to predict the engine behavior with sufficient accuracy [BJP$^+$20]. In our approach this failure of smooth models is turned into a feature, by utilizing a strong divergence between model-predicted and actual performance as an indicator for nonsmooth engine behavior and the necessity to increase the sampling density. This incorporation of a smooth technique is complemented by the use of discrete optimization machinery, specifically *mixed integer nonlinear programs*, to select a set of suitable support points for the engine map from the collected data. This results in our second main contribution [BJP$^+$20].

Main Result 2. *A practical algorithm for the calibration of internal combustion engines in accordance with several EURO emission norms.*

Polytopes and *polyhedra* are an object of study in numerous optimization contexts, e.g., as feasible sets of linear programs or linear regions of piecewise linear functions. The Clarke derivative, another generalized derivative concept, is the convex hull of the matrices that define the linear pieces of the Bouligand derivative [Sch12, p. 89]. For example, for the piecewise linear function corresponding to an AVE the Clarke derivative at the origin of $\mathbb{R}^n$ is a cross polytope with vertices $\mathbb{I} - AS$, where $A \in \mathrm{M}_n(\mathbb{R})$ is the AVE's coefficient matrix and $S \in \operatorname{diag}(\{-1, 1\}^n)$.

The combinatorics of a polytope are uniquely defined by its *face lattice* [Zie95, p. 55]. Above dimension 3 it is not clear if and when a given lattice is the face lattice of a polytope. In dimension 3 this question is settled by *Steinitz' theorem* which says that a graph is the edge graph of a 3-polytope if and only if it is 3-connected and planar [Ste22]. This gives a combinatorial characterization of edge graphs of 3-polytopes.

However, relevant polytope classes can be hybrids of combinatorial and metric restrictions. For example, 3-*prismatoids* are defined partially in combinatorial terms, in that they have two designated facets so that all edges which are not contained in either must be incident to one vertex of each, and partially in metric terms, since the two designated facets must lie in parallel planes.

It is an open question, whether every 3-prismatoid is *edge-unfoldable*, that is, if it can be flattened into the plane without self-overlap after cutting a suitable spanning tree of its edge-graph. The combinatorial structure of 3-prismatoids implies that they can be decomposed into three distinct parts, the two aforementioned designated facets and the connected set of all other facets which is called the *band*. We consider a subclass of 3-prismatoids with an additional metric restriction, called *nested* 3-*prismatoids*. We show that the band of a nested prismatoid can always be cut into two parts which are of “small enough” curvature so that the two band pieces and two designated facets can be, via three carefully selected edge-gluings, reconnected into a surface whose unfolding does not self-intersect [Rad21].

Main Result 3. *A constructive proof that every nested* 3*-prismatoid is edge-unfoldable.*

We will now introduce the three included works in more detail.

0.1 Degree theory for the absolute value equation

The *Linear complementarity problem* (LCP) stands at the crossroads of numerous optimization contexts. Not only do many problems in computational mechanics arise naturally in LCP form. Linear and quadratic programs are special cases of the LCP. For a comprehensive introduction to the topic, see [CPS92]. Other relevant optimization problems, such as equilibrium computations in bimatrix games [CPS92], or – as mentioned above – the solution of arbitrary finite *piecewise linear systems* [GBRS15], can be reduced to solving an LCP.

The by now classical standard form was introduced by Cottle and Dantzig in [CD68]. Let $q \in \mathbb{R}^n$ and $M \in \mathrm{M}_n(\mathbb{R})$, where $\mathrm{M}_n(\mathbb{R})$ denotes the space of $n \times n$ real matrices. Then the linear complementarity problem $\mathrm{LCP}(M, q)$ is to find vectors $v, w \in \mathbb{R}^n_{\geq 0}$ with $w^T v = 0$ so that

$$w \;=\; Mv + q\,.$$

Equivalent systems have been formulated, e.g., via max or min expressions [BC08], and absolute values [Neu90, Chap. 6]. Arguably, the so-called *absolute value equation* (AVE) stands out among these, both in terms of depth and quantity of the associated publications. Let $A \in \mathrm{M}_n(\mathbb{R})$ and $b \in \mathbb{R}^n$. Then the AVE poses the problem to find a vector $z \in \mathbb{R}^n$ so that

$$z - A|z| \;=\; b\,,$$

where $|\cdot|$ denotes the componentwise absolute value. The term absolute value equation was coined only recently by Mangasarian in [Man07], but the first journal publication to investigate the system – in the context of the inversion of interval matrices and the solution of linear interval systems – was authored by Rohn several decades earlier [Roh89]. The equivalence of AVE and LCP was already noted in the latter reference and exploited in an existence proof for LCP solutions that avoids any use of P-matrices.

Via the identities

$$z = \max(0, z) - \max(0, -z) \quad \text{and} \quad |z| = \max(0, z) + \max(0, -z)\,,$$

the AVE can be rewritten as

$$(\mathbb{I} - A)\max(0, z) + (\mathbb{I} + A)\max(0, -z) = b\,.$$

A multiplication with the inverse of either $\mathbb{I} - A$ or $\mathbb{I} + A$, if they exist, turns the above equation into an LCP in standard form.

For for both AVE and LCP unique solvability is comrehensively characterized. The $\mathrm{LCP}(M, q)$ is uniquely solvable for arbitrary right-hand sides $q \in \mathbb{R}^n$ if and only if M is a P-matrix, that is, if all principal minors of M are positive [CPS92]. Let $\mathcal{S}_n$ be the set of $n \times n$ diagonal matrices with entries in $\{-1, 1\}$, then the *sign-real spectrum* of A is defined as the set of real eigenvalues of the 2^n matrices SA, or equivalently AS, where $S \in \mathcal{S}_n$. The largest element of the sign-real spectrum of A is called the *sign-real spectral radius* of A. Rump and Rohn independently showed that the *piecewise linear function*

$$F_A:\; \mathbb{R}^n \to \mathbb{R}^n\,, \quad z \;\mapsto\; z - A|z|$$

is a PL homeomorphism and the AVE uniquely solvable if and only if the sign-real spectral radius is smaller than 1 [Rum97], [Neu90, Chap. 6]. In this case $(\mathbb{I} - A)$ and

$(\mathbb{I}+A)$ are invertible and $(\mathbb{I}-A)^{-1}(\mathbb{I}+A)$, respectively $(\mathbb{I}+A)^{-1}(\mathbb{I}-A)$, the coefficient matrices of the equivalent LCPs in standard form, are P-matrices.

The sign-real spectral radius can be interpreted as a piecewise linear analogue of contractivity conditions for linear operators, or as a generalized Perron root for matrices without sign-restrictions [Rum97]. Its computation is equivalent to the computation of the weighted componentwise distance to the nearest singular matrix [Rum97]. This relation provides a direct connection between the condition of the matrices $\mathbb{I}-AS$ (and $\mathbb{I}-SA$) and the complexity of the AVE, which is NP-complete in general [Chu89], but lies in $\mathrm{O}(n^3)$ for certain uniquely solvable systems with well-conditioned matrices $\mathbb{I}-AS$ [Rad16]. The latter fact makes the AVE particularly interesting in light of recent developments in real algebraic geometry that deal with precisely such connections of complexity and condition, see [AL17, BC13] for founding texts of the research area.

For mere (possibly non-unique) solvability of AVE and LCP there has never been derived a similarly comprehensive characterization as for unique solvability, neither in terms of the sign-real spectrum of A, nor of subdeterminants of the coefficient matrix of an LCP. We close this gap in the theory by characterizing the *mapping degree* of F_A and its LCP-counterpart.

To this end, we define the concept of *aligned values* and the *aligned spectrum.* On the AVE side an aligned value is a nonnegative element of the sign-real spectrum of A so that there exists a corresponding eigenvector of SA in the positive orthant of $\mathbb{R}^n$, or equivalently an eigenvector of AS which lies in the orthant corresponding to the signature S.

We establish that a generic matrix form, which excludes certain degeneracies, can be achieved by a random perturbation with probability 1 (Proposition 1.4.2). Such a perturbation does not affect the mapping degree. This allows us to prove two theorems. The first one gives the following concise formula for the degree mod 2 [RTC19].

Theorem 1.1.1. *Let $A \in \mathbb{R}^{n\times n}$ be generic. Then*

$$\deg F_A \ \equiv (k+1) \mod 2\,,$$

where k is the number of aligning values larger than 1. Moreover, the degree of F_A is 1 if all aligning values are smaller than 1, and 0 if all are larger than 1.

Let $\lambda_1, \lambda_2, \ldots$ be the aligning values of F_A ordered by descending magnitude and let $t \in \mathbb{R}_{\geq 0}$. The key idea of the proof of Theorem 1.1.1 is to establish the degree changes via an analysis of the *homotopy* $(z,t) \mapsto F_{tA}$ in small neighborhoods of the reciprocals of the λ_i by means which are, if not exactly Morse-theoretical, certainly unthinkable without the general inspiration of Morse theory.

The second theorem gives an exact, but more technical formula, for the statement of which we refer to Chapter 1. Let S_i be the signature matrix corresponding to an aligned value λ_i. The exact formula for the degree is derived by summing over the signs of the derivatives of the characteristic polynomials of the matrices AS_i evaluated in the λ_i.

Further, we develop LCP-analogues of all definitions and results that we derived for the AVE. The key reason that the main statements are proved for the lesser known AVE and then transferred to the LCP and not the other way around, is that the structural similarity of the AVE's left side

$$z - A|z| \,=\, \mathbb{I}z - A|z| \,=\, (\mathbb{I}-AS)z \,=\, -(AS-I)z$$

to an eigenvalue equation, which makes it possible to formulate various key arguments in a slicker fashion than it could be done in the LCP-setting.

Concerning complexity, it is known that computation of the sign-real spectral radius is NP-hard. Moreover, determining an aligned vector to a given aligned value λ means to obtain a nontrivial solution for the AVE $z - \lambda^{-1}A = 0$, which is an NP-complete problem due to the NP-completeness of the equivalent problem for the LCP [Chu89]. It is thus likely that determining even a part of the aligned spectrum is a hard problem as well.

0.2 Calibration of internal combustion engines

The *calibration of internal combustion engines* is a problem that has risen to prominence during the so-called diesel crisis. Engine behavior can be modeled as a function from the space of engine *actuators* to the space of engine performance *measurands.* A *calibration solution* is a set of lookup tables, one per actuator. These tables associate an actuator setting to each element of a set of $k \times k$ revolution frequency/torque demands for the engine. Via their combination, the calibration solution associates to each of the $k \times k$ revolution frequency/torque demands a point in the *actuator space.* Values in between are linearly interpolated by the *engine control unit.* A solution is *feasible* if it satisfies two criteria. It must conform to *emission standards*, such as the EURO norms set by the EU, cf. [M$^+$16], and it must be *drivable.* Drivability means that the distance between neighboring solution points in the actuator space is bounded in order to prevent engine damage due to an overly rapid change of settings. An optimal calibration solution is a feasible solution that minimizes fuel consumption.

Traditionally, calibration solutions were obtained by measuring the actuator space on a *uniform grid* and then optimizing on the set of measurements. Modern engines have in the order of ten actuators and sensors each. This leads to a combinatorial explosion of the number of measurements that would have to be performed on a uniform grid. Due to the aforementioned limits of the actuator variation speed, the required number of measurements makes the uniform grid approach infeasible on actual physical test engines. Moreover, even if the measurements could somehow be performed, optimizing over the immense resulting data set would be virtually impossible.

The strategies to circumvent the curse of high dimensionality can be divided into two major lines of thought: *modeling* of the engine behavior via *smooth* functions and *adaptive meshing.* The first approach employs (a comparably small number of) engine measurements to determine the parameters of some model function that is fitted to the engine behavior. Several software packages are available to this end, cf. [KPF$^+$03], [KKL10], and [Mat18]. Due to their smoothness, modeling functions fail to capture the small subsets of the actuator space whereon the engine function displays a "*strongly nonlinear*" behavior, which is a main cause of emission spikes. In an engineering context the term "strong nonlinearity" has to be understood phenomenologically. A function is strongly nonlinear if measurements in some small neighborhood vary extensively in a manner that is not satisfactorily captured by smooth models. Likely underlying causes include insufficient and lack of differentiability. Adaptive meshing methods aim to remedy this problem by concentrating measurements in and near areas of strong nonlinearity.

The starting point of our project was the fact that even the–at the time–most successful adaptive meshing method in the field, the local linear neuro-fuzzy model `LOLIMOT` (local linear model tree) [SHI00], failed to produce calibration solutions for current emission norms. We base our calibration algorithm on `LOLIMOT` and refine the latter in several key aspects. `LOLIMOT` detects insufficient smoothness by comparing local models to actual measurements. This knowledge is used to fit models to smaller and smaller neighborhoods, which may lead to overfitting. We zoom in on strong nonlinearities with a similar detection method. Our algorithm differs from `LOLIMOT` in that we do not produce a (local linear neuro-fuzzy) model, but instead use a *mixed integer nonlinear programming* approach to select actual measured *data points* for the final calibration solution.

Further, `LOLIMOT` composes the calibration solution componentwise, which can lead to redundancies during the physical experiments and unpredictable interactions of the component functions during the assembly of the calibration solution. Our process considers all actuators and measurands at once by a randomized measurement distribution approach for the whole actuator space which is weighted by the density of measurements already taken and the measure of local smoothness described above. This combination of methods allows us to significantly improve on the performance of `LOLIMOT`. The main result of our work [BJP$^+$20] is that we manage to calibrate a benchmark engine model by AVL [Ve13] for several EURO emission norms.

0.3 Edge-unfolding nested prismatoids

The question whether any 3-*polytope* is *edge-unfoldable*, that is, whether it is possible to cut its boundary along some spanning tree of its edge graph so that the resulting connected surface may be unfolded flat into the plane without self-overlaps, can be dated back to the “Painter’s Manual” by Albrecht Dürer [Dü25]. It is thus often referred to as *Dürer’s problem.* The first author to explicitly state it was Shephard [She75]. Grünbaum conjectured the question to have a positive answer [Grü91]. His conjecture is commonly referred to as *Dürer’s conjecture.* Its status is still open.

Several related problems were solved. For dimensions $n \geq 4$ the analogous question to Dürer’s conjecture, is it possible to cut the boundary of a convex d-polytope along its ridges without disconnecting it so that the resulting $(d-1)$-dimensional surface can be isometrically embedded into a hyperplane of $\mathbb{R}^d$, has been positively answered [MP08]. Another problem concerns edge-unfoldability of non-convex polytopes which are combinatorially equivalent to a convex 3-polytope. Several ununfoldable families of such polytopes are known, cf. [Grü02, Tar99, DDE20].

Moreover, any set of cuts into the surface of a 3-polytope that forms a tree which spans the vertices will yield a surface that can be immersed isometrically into a plane. Two cutting strategies based on this general requirement, which are in some sense dual, have been shown to yield overlap-free unfoldings which are called *star* and *source* unfoldings. The star unfolding is obtained by picking a “*generic*” point x in a convex 3-polytope P and then cutting the shortest paths (in the intrinsic metric of P) between x and the vertices of P. By “generic” we mean that x is picked so that the shortest paths to the vertices are unique and do not intersect more than one vertex, which is the case for almost all $x \in P$. The star unfolding was introduced by Alexandrow [Ale50, p. 195] and proved to lead to an overlap-free unfolding by Aronov and O’Rourke [AO92]. For the source unfolding, P is cut along the so-called cut locus, which is the closure of the set of points with a non-unique shortest path to a source point x in P. The cut locus was introduced by Poincaré [Poi05]. The resulting unfolding trivially is an embedding because unfoldings are isometries which means that shortest paths in P unfold into straight lines in the plane.

Concerning the original problem, there are indications both for a positive and a negative answer. Ghomi showed that every 3-polytope can be unfolded after an affine stretching, which implies that there are no combinatorial obstructions to edge-unfoldability [Gho14]. O'Rourke showed that any sufficiently flat, acutely triangulated *polyhedral cap* is edge-unfoldable [O'R18], [O'R17]. Also, large scale computer experiments have found overlap-free unfoldings for every tested polytope [Sch97]. On the other hand, similar experiments revealed that the probability that a randomly picked spanning tree of the edge graph leads to an overlap-free unfolding goes to zero with increasing complexity of the polytope [Sch90]. Moreover, a generalized form of Dürer's conjecture concerning so-called pseudo-edges, which are certain geodesics in the intrinsic metric of the polytope, has recently been falsified [BG17]. Finally, Dürer's conjecture could only be verified for very narrow classes of polytopes.

Among these classes are *domes* and *prismoids*. A dome is a 3-polytope that has one designated facet to which all others are incident. There exist several proofs of the edge-unfoldability of domes, for example in the book on unfolding algorithms by Demaine and O'Rourke [DO07]. A *prismatoid* is the convex hull of two convex polygons A, B in parallel planes H_A and H_B, respectively. A *prismoid* is a prismatoid whose lateral facets are trapezoids. O'Rourke proved the edge-unfoldability of prismoids via the so-called *petal unfolding* strategy, where the band is either cut once at every vertex of A, or once at every vertex of B [O'R01].

It is not known whether a general prismatoid is unfoldable. Prismatoids are significantly more complicated than prismoids, because the lateral facets can be trapezoids, triangles which contain an edge of A, and triangles which contain an edge of B in arbitrary sequence. A prismatoid is *nested* if the orthogonal projection of A to H_B is properly contained in B, or vice versa. The main result of this thesis' final section is [Rad21]:

Theorem 3.1.1. *Any nested prismatoid is edge-unfoldable.*

Nested prismatoids are polyhedral caps. This observation allows us to follow a strategy in the proof which was pioneered by O'Rourke in his work on nearly flat polyhedral caps, and can be summarized as "flatten, cut, project back up, unfold". The adaptation of this approach to the case of nested prismatoids relies on two gimmicks. First, we diverge from the two standard methods to unfold prismatoids, which are the *band unfolding*, where the band is unfolded in one piece, and the above-mentioned petal unfolding. Instead, we cut the band (of an arbitrary nested prismatoid P) into two pieces whose curvature we bound via a careful selection of the cut edges. Second, we either attach trapezoids to the ends of the two band pieces or embed them in trapezoids, depending on the situation. We then prove that A, B and the two extended – but easier to analyze – band pieces can be glued together along three suitable edges so that the resulting polyhedral surface can be unfolded without self-overlap, which implies edge-unfoldability of P. A frequently used tool in our analysis is the notion of radially monotone polygonal paths introduced by O'Rourke in [O'R18].

Chapter 1

Degree theory for the absolute value equation

This chapter is taken from the article "Generalized Perron roots and solvability of the absolute value equation" [RTC19] by Manuel Radons and Josué Tonelli-Cueto. An extended abstract of this work has been published in the proceedings of the Discrete Mathematics Days 2022 [RTC22].

1.1 Introduction

The *linear complementarity problem* $\mathrm{LCP}(q, M)$, where $q \in \mathbb{R}^n$, and $M \in \mathrm{M}_n(\mathbb{R})$, the space of $n \times n$ real matrices, is to determine $v, w \in \mathbb{R}^n_{\geq 0}$ with $v^T w = 0$ so that

$$v \;=\; Mw + q\,. \tag{1.1}$$

It provides a common framework for numerous optimization tasks in economics, engineering and computer science. Classical problems that can be reduced to solving an LCP include bimatrix games, linear and quadratic programs [CPS92]. Recent applications are the correct formulation of numerical models for free-surface hydrodynamics [BC08], L1 regularization in reinforcement learning [JPwP10], and the massively parallel implementation of collision detection on CUDA GPUs [Ngu07, Chap. 33].

It is well known that an $\mathrm{LCP}(q, M)$ is uniquely solvable for arbitrary q if and only if M is a P-matrix, that is, a matrix whose principal minors are all positive. If $\mathrm{LCP}(q, M)$ is solvable—possibly non-uniquely—for arbitrary q, then M is called a *Q-matrix.* There exists no comprehensive characterization of Q-matrices. We will study this question by investigating the following equivalent problem [MM06]. Let $b \in \mathbb{R}^n$ and $A \in \mathrm{M}_n(\mathbb{R})$. Then the *absolute value equation* (AVE) poses the problem to find a vector $z \in \mathbb{R}^n$ so that

$$z - A|z| \;=\; b\,, \tag{1.2}$$

where $|\cdot|$ denotes the componentwise absolute value. The AVE is an interesting problem in its own right. For example, a result by Rump [Rum97, Thm. 2.8] relates the number of solutions of (1.2) to the condition of the matrix A, which is noteworthy in light of recent developments in real algebraic geometry that deal with precisely such connections of complexity and condition [BC13, Part III]. However, the main focus of theoretical investigations of the AVE is to obtain statements about the LCP. A recent success of this approach is the development of condition numbers for the AVE that lead to new error bounds for the LCP [ZH22].

We will study solvability of the AVE, but with our eyes on the Q-matrix problem. To this end we investigate the piecewise linear function

$$\begin{aligned} F_A : \mathbb{R}^n &\to \mathbb{R}^n \\ z &\mapsto z - A|z| \end{aligned} \tag{1.3}$$

associated to the AVE (1.2) and determine its degree and its degree modulo 2 (see Section 1.3.1) in terms of the *aligned spectrum* of A (see Section 1.2):

$$\mathrm{Spec}^{\mathrm{a}}(A) := \{\lambda \geq 0 \mid \exists x \neq 0 \,:\, |Ax| = \lambda x\}. \tag{1.4}$$

The first main result of this article relates the degree of F_A to what we call the *aligned count* of A:

$$\mathrm{c}^{\mathrm{a}}(A) := \#\{\lambda \in \mathrm{Spec}^{\mathrm{a}}(A) \mid \lambda > 1\} \tag{1.5}$$

where the count on the right-hand side is with multiplicities.

The term *generic* in the theorem and the corollaries means that the statement holds for all matrices that have a specific property (see Definition 1.4.1), which is satisfied for all matrices except those in a given homogeneous hypersurface (see Section 1.4). This condition, akin to the general position condition in the polyhedral world, guarantees that a random matrix (with respect to a continuous distribution) is generic with probability 1.

Theorem 1.1.1. *Let* $A \in \mathrm{M}_n(\mathbb{R})$ *be generic such that* $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$. *Then the degree of* F_A *is well-defined and it satisfies that*

$$\deg F_A \;\equiv\; 1 + \mathrm{c}^{\mathrm{a}}(A) \mod 2 \tag{1.6}$$

Moreover, $\deg F_A$ *equals* 1 *if all aligned values are smaller than* 1*, and it equals* 0 *if all aligned values are larger than* 1.

Corollary 1.1.2. *Let* $A \in \mathrm{M}_n(\mathbb{R})$ *be a generic matrix. Then the number of aligned values of* A*, counted with multiplicity, is odd.*

Corollary 1.1.3. *Let* $A \in \mathrm{M}_n(\mathbb{R})$ *be a generic matrix such that* $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$. *If* $\mathrm{c}^{\mathrm{a}}(A)$ *is even, then the AVE* (1.2) *has a solution for every* $b \in \mathbb{R}^n$.

After stating Theorem 1.1.1, we might wonder if there is an exact formula for the degree of F_A when A is generic (in the sense of Definition 1.4.1). Indeed, there is such formula and Theorem 1.1.1 is a direct consequence of the following more general—but more technical—theorem.

Theorem 1.1.4. *Let* $A \in \mathrm{M}_n(\mathbb{R})$ *be generic and such that* $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$. *Then the degree of* F_A *is well-defined and it satisfies that*

$$\deg F_A = 1 - \sum\{\mathrm{sign}(\chi'_{SA}(\lambda)) \mid \lambda > 1, S \in \mathcal{S}, \exists x \in \mathbb{R}_{>0} \,:\, SAx = \lambda x\}$$

where χ_{SA} *is the characteristic polynomial of* SA*, and* $\mathcal{S} \subseteq \mathrm{M}_n(\mathbb{R})$ *is the set of sign matrices, i.e., diagonal matrices with* ± 1 *in the diagonal entries.*

Observe that the right-hand side sum runs over all aligned values greater than 1, since we have that $|Ax| = \lambda x$ for some $x \neq 0$ if and only if $SAx = x$ for some $S \in \mathcal{S}$ and $x \in \mathbb{R}^n_{\geq 0}$. Now, for a generic matrix (see Definition 1.4.1), all aligned values correspond to simple eigenvalues of some SA, and so the right-hand side sum is nothing more than a "signed aligned count", i.e., a signed variation of the aligned count $\mathrm{c}^{\mathrm{a}}(A)$. In this way, Theorem 1.1.1 is just Theorem 1.1.4 reduced modulo 2.

As we mentioned above, a key reason to study the AVE is to gain insights into the equivalent LCP. To this end we derive LCP-analogues for the concept of the aligned spectrum and all statements about the degree of F_A listed in this introduction. Concerning our afore-stated interest in Q-matrices, we note that Corollary 1.1.3 directly translates into a statement about the latter, i.e., the coefficient matrix of an LCP is a Q-matrix if the LCP-equivalent of the aligned count is even (Corollary 1.5.7).

Organization In Section 1.2, we recall the sign-real spectrum and we introduce the aligned spectrum which naturally emerges during the study of the eigenproblem of F_A; in Section 1.3, we recall the topological notion of degree, its formula in terms of signed counts of preimages of a regular value, and some of its properties in the case of interest; in Section 1.4, we introduce the notion of genericity of a matrix relevant to our context and show some perturbation results. In Section 1.5, we show how the above results for AVEs can be transferred to LCPs. We conclude in Section 1.6 proving Theorems 1.1.1 and its corollaries, and also Theorem 1.1.4.

1.2 Sign-Real and Aligned Spectra

The sign-real and aligned spectra of A emerge naturally when we study the map F_A (1.3). Note that studying this map in order to understand (1.2) is a similar to the strategy in linear algebra to study $z \mapsto Az$ in order to understand the solvability of $Az = b$

We note that F_A is a positively homogeneous map, i.e., for every $z \in \mathbb{R}^n$ and $\lambda > 0$, $F_A(\lambda z) = \lambda F_A(z)$; and that F_A is piecewise linear, with linear parts of the form

$$\mathbb{I} - AS \tag{1.7}$$

for sign matrices $S \in \mathcal{S} := \{\mathrm{diag}(s_1, \ldots, s_n) \mid s_i \in \{-1, +1\}\}$.

1.2.1 Sign-real spectrum and bijectivity

The sign-real spectrum was used independently by Rump [Rum97] and Rohn [Neu90, Chap. 6] to determine when the function F_A is bijective, or equivalently, when the absolute value equation (1.2) is uniquely solvable for arbitrary b.

Definition 1.2.1. *Let $A \in \mathrm{M}_n(\mathbb{R})$. The* sign-real spectrum *of A, $\mathrm{Spec}^{\rho}(A)$, is the set*

$$\mathrm{Spec}^{\rho}(A) := \mathbb{R}_{\geq 0} \cap \bigcup_{S \in \mathcal{S}} \mathrm{Spec}(SA).$$

Theorem 1.2.2. *Let $A \in \mathrm{M}_n(\mathbb{R})$. Then the following are equivalent:*

(a) $\max \mathrm{Spec}^{\rho}(A) < 1$,

(b) F_A *is bijective,*

(c) The AVE (1.2) *has a unique solution for every* $b \in \mathbb{R}^n$. ☐

The largest element of the sign-real spectrum, $\max \operatorname{Spec}^{\rho}(A)$, is called the *sign-real spectral radius*. Due to Theorem 1.2.2 it can be considered as a generalization of contractivity conditions for linear operators. Rump [Rum97] also showed that the sign-real spectral radius generalizes the Perron root to matrices that are not positive. Another generalization of Perron Frobenius theory, to homogeneous monotone functions on the positive cone, was introduced in [GG04]. It is not clear how these two generalized theories are related apart from their common origin.

1.2.2 Aligned spectrum and the eigenproblem

The aligned spectrum arises when studying the eigenproblem for F_A: Determine for which $\lambda \geq 0$ and $v \in \mathbb{R}^n \setminus 0$, we have

$$F_A(v) = \lambda v.$$

Proposition 1.2.3. *Let* $A \in \mathrm{M}_n(\mathbb{R})$, $\lambda \geq 0$ *and* $v \in \mathbb{R}^n \setminus 0$. *Then* $F_A(v) = \lambda v$ *if and only if there is some sign matrix* $S \in \mathcal{S}$ *such that* $Sv \geq 0$ *and* $(1-\lambda, |v|)$ *is an eigenpair of* SA.

Proof. If $F_A(v) = \lambda v$, then we have that $(1-\lambda)v = A|v|$. Now, let $S \in \mathcal{S}$ such that $Sv \geq 0$, by taking as the diagonal elements of S the signs of the components of v. Then $(1-\lambda)|v| = (1-\lambda)Sv = SA|v|$. Hence there is $S \in \mathcal{S}$ such that $Sv \geq 0$ and $(1-\lambda, |v|)$ is an eigenpair of SA.

Conversely, if such an S exists, then

$$F_A(v) = v - A|v| = v - S(SA)|v| = v - S(1-\lambda)|v| = \lambda v,$$

where we have used that $S^2 = \mathbb{I}$. Consequently, $(1-\lambda, |v|)$ is an eigenpair of SA, and $v = S|v|$. ☐

In view of the above proposition, the aligned spectrum is introduced. We note that the definition below is equivalent to that given in (1.4).

Definition 1.2.4. *Let* $A \in \mathrm{M}_n(\mathbb{R})$. *An* aligned trio *of* A *is a triplet* $(\lambda, S, v) \in \mathbb{R}_{\geq 0} \times \mathcal{S} \times (\mathbb{S}^{n-1} \cap \mathbb{R}^n_{\geq 0})$ *such that*

$$SAv = \lambda v.$$

Given such a trio, we call (S, v) *an* aligned vector *and* λ *an* aligned value *of* A. *The* aligned spectrum *of* A, *which we denote* $\operatorname{Spec}^{\mathrm{a}}(A)$, *is the set of aligned values of* A, *i.e.,*

$$\operatorname{Spec}^{\mathrm{a}}(A) := \{\lambda \geq 0 \mid \exists S \in \mathcal{S},\ v \in \mathbb{R}^n_{\geq 0} \setminus 0\ :\ SAv = \lambda v\}.$$

The following proposition shows how the aligned spectrum is related to the solution set of $F_A(z) = 0$. In analogy to linear maps, we call F_A *nondegenrate* if $F_A(z) = 0$ has only the trivial solution $z = 0$.

Proposition 1.2.5. *Let* $A \in \mathrm{M}_n(\mathbb{R})$*. Then* $F_A(z) = 0$ *has non-trivial solutions if and only if* $1 \in \mathrm{Spec}^{\mathrm{a}}(A)$*.*

Proof. By Proposition 1.2.3, non-trivial solutions of $F_A(z) = 0$ correspond to aligned trios of the form $(1, S, v)$. Hence, the claim follows. $\square$

From the definitions it follows that

$$\mathrm{Spec}^{\mathrm{a}}(A) \subseteq \mathrm{Spec}^{\rho}(A). \tag{1.8}$$

However, this is not an equality in general.

Example 1.2.6. *Let*

$$A := \begin{pmatrix} 1 & 0 \\ -1/2 & 1/2 \end{pmatrix}. \tag{1.9}$$

One readily checks that

$$\mathrm{Spec}^{\mathrm{a}}(A) = \{1/2\} \subsetneq \mathrm{Spec}^{\rho}(A) = \{1/2, 1\}.$$

Moreover, by Theorem 1.2.2 and Proposition 1.2.5, this example shows that F_A *might not be bijective, even though it is nondegenrate. Furthermore, it shows that the largest aligned value and the sign-real spectral radius do not necessarily coincide. In light of Theorem 1.1.1, this demonstrates that we may have* $\deg F_A = 1$ *without bijectivity.*

We finish with the following example which shows that $A \mapsto \max \mathrm{Spec}^{\mathrm{a}}(A)$ is not continuous unlike $A \mapsto \max \mathrm{Spec}^{\rho}(A)$ which is [Rum97, Corollary 2.5]. However, in the generic case (see Definition 1.4.1), we can recover continuity, since simple real eigenvalues cannot become complex and strictly positive vectors cannot become nonpositive under an arbitrarily small perturbation.

Example 1.2.7. *Let* t *lie in a sufficiently small neighborhood of* 0 *and consider the following family of matrices:*

$$A_t := \begin{pmatrix} 1 & -0.5 - t \\ 0.5 & 0 \end{pmatrix}. \tag{1.10}$$

A straightforward calculation shows that $\mathrm{Spec}^{\rho}(A_t)$ *is equal to*

$$\left\{ \frac{1 + \sqrt{-2t}}{2}, \frac{1 - \sqrt{-2t}}{2}, \frac{\sqrt{2}\sqrt{1+t} - 1}{2}, \frac{1 + \sqrt{2}\sqrt{1+t}}{2} \right\},$$

if $t \le 0$*; and*

$$\left\{ \frac{\sqrt{2}\sqrt{1+t} - 1}{2}, \frac{1 + \sqrt{2}\sqrt{1+t}}{2} \right\}$$

if $t > 0$*. Similarly, we can see that* $\mathrm{Spec}^{\mathrm{a}}(A_t)$ *is equal to*

$$\left\{ \frac{1 + \sqrt{-2t}}{2}, \frac{1 - \sqrt{-2t}}{2}, \frac{\sqrt{2}\sqrt{1+t} - 1}{2} \right\},$$

if $t \le 0$*; and*

$$\left\{ \frac{\sqrt{2}\sqrt{1+t} - 1}{2} \right\}$$

if $t > 0$.

Hence, we have that

$$\max \mathrm{Spec}^{\rho}(A_t) \;=\; \frac{1+\sqrt{2}\sqrt{1+t}}{2} \;\geq\; \max \mathrm{Spec}^{\mathrm{a}}(A_t) \;=\; \begin{cases} \frac{1+\sqrt{-2t}}{2}\,, & \text{if } t \leq 0\,, \\ \frac{\sqrt{2}\sqrt{1+t}-1}{2}\,, & \text{if } t > 0\,. \end{cases}$$

This shows that the maximum of the aligned spectrum is not continuous.

1.3 Degree of a map

The degree of a continuous map $G : \mathbb{S}^{n-1} \to \mathbb{S}^{n-1}$ is a fundamental topological invariant that is preserved under homotopy. Intutively, we only have to think of the degree as the number of times that the map wraps $\mathbb{S}^{n-1}$ around itself. Its formal definition is as follows.

Definition 1.3.1. *Let* $G : \mathbb{S}^{n-1} \to \mathbb{S}^{n-1}$ *be a continuous map. The* degree *of* G, $\deg G$, *is the unique integer* d *such that the induced map* $H_{n-1}(f) : H_{n-1}(\mathbb{S}^{n-1}) \to H_{n-1}(\mathbb{S}^{n-1})$ *of homology groups is given by*

$$x \mapsto dx$$

under the choice of any fixed isomorphism $H_{n-1}(\mathbb{S}^{n-1}) \simeq \mathbb{Z}$.

Among the main properties of the degree, we have the following, cf. [OR09, p. 98 ff].

Proposition 1.3.2. *Let* $G_0, G_1 : \mathbb{S}^{n-1} \to \mathbb{S}^{n-1}$ *be continuous maps. Then the following holds:*

(1) $\deg \mathrm{id}_{\mathbb{S}^{n-1}} = 1$ *and* $\deg(-\mathrm{id}_{\mathbb{S}^{n-1}}) = (-1)^n$.

(2) $\deg G_1 \circ G_0 = \deg G_1 \deg G_0$.

(3) If there is a homotopy *between* G_0 *and* G_1, *that is, a continuous map* $H : [0,1] \times \mathbb{S}^{n-1} \to \mathbb{S}^{n-1}$ *such that for all* $x \in \mathbb{S}^{n-1}$, $H(0,x) = G_0(x)$ *and* $H(1,x) = G_1(x)$, *then*

$$\deg G_0 = \deg G_1.$$

Moreover, the converse statement is also true.

(4) If G_0 *is not surjective, then* $\deg G_0 = 0$. ☐

Our investigation will be centered around F_A which are nondegenerate. In this case, we can consider the spherical map

$$\begin{aligned} \bar{F}_A : \mathbb{S}^{n-1} &\to \mathbb{S}^{n-1} \\ x &\mapsto F_A(x)/\|F_A(x)\|_2 \end{aligned} \tag{1.11}$$

and define the *degree of* F_A as

$$\deg F_A := \deg \bar{F}_A. \tag{1.12}$$

This definition agrees with a more traditional count used for maps $\mathbb{R}^n \mapsto \mathbb{R}^n$. Recall that the *set of regular values of* F_A is the set given by

$$\mathrm{Reg} F_A := \{y \in \mathbb{R}^n \mid \forall x \in F_A^{-1}(y),\, \partial F_A(x) \text{ is well-defined and invertible}\},$$

where ∂F_A is the Jacobian of F_A.

Proposition 1.3.3. *Let $A \in \mathrm{M}_n(\mathbb{R})$ be such that $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$. Then the set of regular values of F_A is dense and for all $y \in \mathrm{Reg}F_A$ we have*

$$\deg F_A = \sum_{x \in F_A^{-1}(y)} \mathrm{sign}(\det(\partial F_A(x))). \tag{1.13}$$

Proof. By [CPS92, p. 509 ff] the oriented preimage counts of a nondegenerate positively homogeneous function and its restriction to the sphere coincide. ☐

Moreover, the following proposition is helpful for our degree computations.

Proposition 1.3.4. *Let $A, A_0, A_1 \in \mathrm{M}_n(\mathbb{R})$. Then:*

(1) If $A : [0,1] \to \mathrm{M}_n(\mathbb{R})$ is a continuous path between A_0 and A_1 such that for all $t \in [0,1]$, $1 \notin \mathrm{Spec}^{\mathrm{a}}(A(t))$; then

$$\deg F_{A_0} = \deg F_{A_1}.$$

(2) If $\mathrm{Spec}^{\mathrm{a}}(A) \subseteq [0,1)$, then $\deg F_A = 1$.

(3) If $\mathrm{Spec}^{\mathrm{a}}(A) \subset (1,\infty)$, then $\deg F_A = 0$.

Proof. (1) Consider the following homotopy between $\bar{F}_{A_0}$ and $\bar{F}_{A_1}$:

$$[0,1] \times \mathbb{S}^{n-1} \ni (t,x) \mapsto H(t,x) = \bar{F}_{A(t)}(x) = \frac{F_{A(t)}(x)}{\|F_{A(t)}(x)\|_2}.$$

This homotopy is well-defined because, by assumption and Proposition 1.2.5, $F_{A(t)}(x)$ does not vanish at any (t,x). Hence $\bar{F}_{A_0}$ and $\bar{F}_{A_1}$ are homotopic and so they have the same degree.

(2) Consider the path

$$[0,1] \ni t \mapsto A(t) := (1-t)A.$$

This path joins A with $\mathbb{O}$ and it satisfies the condition of (1). Hence $\deg F_A = \deg F_{\mathbb{O}} = \deg \mathrm{id}_{\mathbb{S}^{n-1}} = 1$.

(3) Consider the following homotopy

$$[0,1] \times \mathbb{S}^{n-1} \ni (t,x) \mapsto H(t,x) = \frac{(1-t)x - A|x|}{\|(1-t)x - A|x|\|_2}.$$

Note that the map $(t,x) \mapsto (1-t)x - A|x|$ is continuous, so if it is not vanishing, then the above homotopy is well-defined and continuous. If $t < 1$, then

$$\mathbb{S}^{n-1} \ni x \mapsto (1-t)x - A|x| = (1-t)F_{A/(1-t)}(x)$$

cannot vanish by Proposition 1.2.5, since $1 \notin \mathrm{Spec}^{\mathrm{a}}(A/(1-t)) = \mathrm{Spec}^{\mathrm{a}}(A)(1-t) \subset (1,\infty)$. If $t = 1$, then

$$\mathbb{S}^{n-1} \ni x \mapsto A|x|$$

does not vanish, because otherwise $0 \notin \mathrm{Spec}^{\mathrm{a}}(A)$. Thus the desired map does not vanish and we obtain a homotopy between F_A and

$$\mathbb{S}^{n-1} \ni x \mapsto \frac{A|x|}{\|A|x|\|_2}.$$

If we precompose this map with $x \mapsto |x|$, it does not change. Now, since $x \mapsto |x|$ is not surjective,

$$\deg\left(x \mapsto \frac{A|x|}{\|A|x|\|_2}\right) = \deg\left(x \mapsto \frac{A|x|}{\|A|x|\|_2}\right)\deg(x \mapsto |x|) = 0,$$

as we wanted to show. □

We conclude with an example which shows that the relationship between the aligned spectrum and the degree in Theorem 1.1.1 holds only modulo 2

Example 1.3.5. *Let ε be in a sufficiently small neigborhood of zero. Consider the family of matrices:*

$$B_\varepsilon := \begin{pmatrix} 2.5 & -1.25-\varepsilon \\ 1.25 & 0 \end{pmatrix}. \tag{1.14}$$

We can see that

$$\mathrm{Spec}^{\mathrm{a}}(B_\varepsilon) = \left\{1.25\left(1+\sqrt{-0.8\varepsilon}\right), 1.25\left(1-\sqrt{-0.8\varepsilon}\right), 1.25\left(\sqrt{2}\sqrt{1+0.4\varepsilon}-1\right)\right\},$$

if $\varepsilon \leq 0$; and that

$$\mathrm{Spec}^{\mathrm{a}}(B_\varepsilon) = \left\{1.25\left(\sqrt{2}\sqrt{1+0.4\varepsilon}-1\right)\right\},$$

if $\varepsilon > 0$. And so we see that

$$\#\{\lambda \in \mathrm{Spec}^{\mathrm{a}}(B_\varepsilon) \mid \lambda > 1\} = \begin{cases} 2, & \text{if } \varepsilon < 0 \\ 0, & \text{if } \varepsilon > 0 \end{cases},$$

and that

$$\deg F_{B_\varepsilon} = 1$$

by Proposition 1.3.4, cf. Figure 1.

On the one hand, this shows that the degree is more stable than the number of aligned values greater than one. On the other hand, note that the change in the number of aligned values greater than one happens because for $\varepsilon = 0$, B_ε is not generic—it has a double aligned value.

Moreover, for $\varepsilon < 0$, B_ε is generic (see Definition 1.4.1) and it satisfies

$$\#\{\lambda \in \mathrm{Spec}^{\mathrm{a}}(B_\varepsilon) \mid \lambda > 1\} - 1 = \deg F_{B_\varepsilon};$$

and for $\varepsilon > 0$, B_ε is still generic, but

$$\#\{\lambda \in \mathrm{Spec}^{\mathrm{a}}(B_\varepsilon) \mid \lambda > 1\} + 1 = \deg F_{B_\varepsilon}.$$

This shows that the equality modulo 2 in Theorem 1.1.1 cannot be corrected in an easy way to obtain an equality between the degree and the number of aligned values greater than one. We need the more technical expression in Theorem 1.1.4 for this.

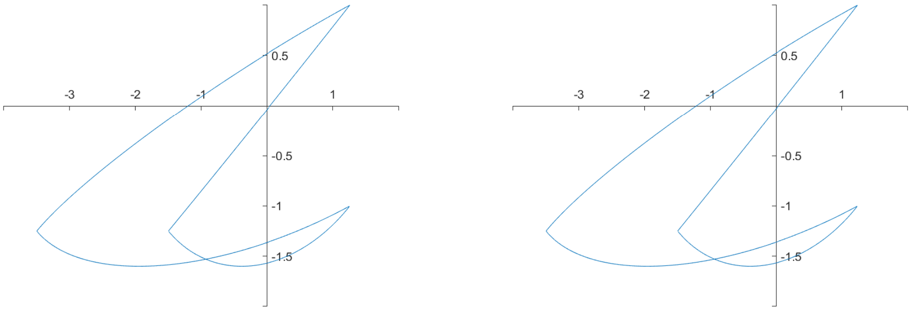


Figure 1: Image of the unit circle under F_{B_ε}. For $\varepsilon = -0.01$ (left) and $\varepsilon = 0.01$ (right) it winds around the origin.

1.4 Generic matrices

To make precise the statement of Theorem 1.1.1, we introduce a notion of genericity adapted to our setting.

Definition 1.4.1. *A* generic *matrix is a matrix $A \in \mathrm{M}_n(\mathbb{R})$ such that for every aligned trio (λ, S, v) of A, a) λ is a simple eigenvalue of SA, and b) v is strictly positive.*

Since genericity usually means a class of entities whose complement is contained in a proper algebraic hypersurface, we need to show that the above notion is indeed a generic one according to the common use.

Proposition 1.4.2. *The set of matrices that is not generic in $\mathrm{M}_n(\mathbb{R})$ is contained in a proper algebraic hypersurface.*

In particular, for any random matrix $\mathfrak{A} \in \mathrm{M}_n(\mathbb{R})$ with an absolutely continuous distribution, $\mathfrak{A}$ is generic almost surely.

Proof. We only have to prove that the set of matrices with double aligned values or aligned vectors in the boundary of the positive orthant is contained in a hypersurface.

The above set is contained in the union of the sets

$$\{A \in \mathrm{M}_n(\mathbb{R}) \mid SA \text{ has a double eigenvalue}\} \tag{1.15}$$

and

$$\{A \in \mathrm{M}_n(\mathbb{R}) \mid SA \text{ has a non-zero eigenvector in } H\} \tag{1.16}$$

where $S \in \mathcal{S}$ runs over all sign matrices and H over all coordinate hyperplanes—of the form $X_i = 0$. Thus, if we show that each one of these sets is contained in a hypersurface, then we are done, since a finite union of hypersurfaces is a hypersurface.

On the one hand, the set 1.15 is given by the discriminant of the characteristic polynomial of SA, which is well-known to define a proper algebraic hypersurface in $\mathrm{M}_n(\mathbb{R})$. On the other hand, the set 1.16 is a proper algebraic hypersurface by [OS13, Propoposition 1.2]. Hence, all the sets are proper algebraic hypersurfaces, and the proof is complete. ☐

1.4.1 Interpretation of count for generic matrices

When we introduced the aligned count, see (1.5),

$$\mathrm{c}^{\mathrm{a}}(A) \;=\; \#\{\lambda \in \mathrm{Spec}^{\mathrm{a}}(A) \mid \lambda > 1\},$$

we said that the right-hand side counts multiplicities. Note that for generic A, an aligned value λ will always be a simple eigenvalue of the corresponding SA, where $S \in \mathcal{S}$. However, there might be more than one such SA. Because of this we have to include the "counted with multiplicity". The following proposition gives an alternative interpretation of the central quantity $\mathrm{c}^{\mathrm{a}}(A)$ for Theorem 1.1.1 in terms of $\bar{F}_A$.

Proposition 1.4.3. *Let $A \in \mathrm{M}_n(\mathbb{R})$ be generic. Then $\mathrm{c}^{\mathrm{a}}(A)$ is equal to the number of fixed points of $\bar{F}_A$ such that its antipodal point is mapped to them. In other words,*

$$\mathrm{c}^{\mathrm{a}}(A) \;=\; \#\{x \in \mathbb{S}^{n-1} \mid x = \bar{F}_A(x) = \bar{F}_A(-x)\}.$$

For proving this proposition, the following proposition will be useful.

Proposition 1.4.4. *Let $A \in \mathrm{M}_n(\mathbb{R})$ and $x \in \mathbb{S}^{n-1}$. If F_A is nondegenerate, then x is a fixed point of $\bar{F}_A$ if and only if there is an aligned trio (λ, S, v) such that either $x = -Sv$ or $\lambda < 1$ and $x = Sv$. Moreover, when x is a fixed point of $\bar{F}_A$, the following are equivalent:*

- $\bar{F}_A(-x) = x$.
- *There is an aligned trio (λ, S, v) such that $\lambda > 1$ and $x = -Sv$.*

Proof of Proposition 1.4.3. By Proposition 1.4.4, the fixed points $x \in \mathbb{S}^{n-1}$ of $\bar{F}_A$ such that $\bar{F}_A(-x) = x$ are in one-to-one correspondence with the aligned trios (λ, S, v) such that $\lambda > 1$. Since A is generic, this means precisely the number of aligned values greater than one counted with multiplicity. ☐

Proof of Proposition 1.4.4. If (λ, S, v) is an aligned trio, then

$$F_A(Sv) = (1 - \lambda)Sv$$

and

$$F_A(-Sv) = (1 + \lambda)(-Sv).$$

In this way, $-Sv$ is always a fixed point of $\bar{F}_A$ and Sv is so if and only if $\lambda < 1$. This shows one direction.

If $x \in \mathbb{S}^{n-1}$ is a fixed point of $\bar{F}_A$, then for some $\mu > 0$,

$$F_A(x) = \mu x.$$

Let $S \in \mathcal{S}$ be such that $Sx \geq 0$, so that $v = Sx$. Then we have that

$$SAv = (1 - \mu)v.$$

If $1 - \mu \geq 0$, then $(1 - \mu, S, v)$ is an aligned trio such that $1 - \mu < 1$ and $x = Sv$. Otherwise, $1 - \mu < 0$, and then $(\mu - 1, -S, v)$ is an aligned trio and $x = -(-S)v$. Hence there is an aligned trio (λ, S, v) such that either $x = -Sv$ or $\lambda < 1$ and $x = Sv$.

We show the second equivalence. Let $x \in \mathbb{S}^{n-1}$ be a fixed point of $\bar{F}_A$. Then, by the first part, there is an aligned trio (λ, S, v) such that either $x = -Sv$ or $\lambda < 1$ and $x = Sv$. In the second case, we have that

$$\bar{F}_A(-x) = -x.$$

Thus we must have the first case. But then $\bar{F}_A(-x) = x$ if and only if $\lambda > 1$, because otherwise $-x$ is also a fixed point.

If $x = -Sv$ for some aligned trio (λ, S, v) such that $\lambda > 1$. Then, by the first equivalence, $x = -Sv$ is a fixed point of $\bar{F}_A$, and, by direct computation, $\bar{F}_A(-x) = x$. ☐

1.4.2 Perturbation of matrices to make them generic

The following proposition shows that matrices corresponding to non-degenerate maps can be slightly perturbed to obtain a generic matrix with the same corresponding degree.

Proposition 1.4.5. *Let* $A \in \mathrm{M}_n(\mathbb{R})$ *be such that* $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$. *Then:*

(a) The quantity

$$\kappa^{\mathrm{a}}(A) := \sup_{x \neq 0} \frac{\|x\|_2}{\|F_A(x)\|_2} \tag{1.17}$$

is finite.

(b) For every $\varepsilon \in (0, 1/\kappa^{\mathrm{a}}(A))$ *and*

$$\tilde{A} \in B_F(A, \varepsilon) := \{X \in \mathrm{M}_n(\mathbb{R}) \mid \|X - A\|_F < \varepsilon\},$$

$F_{\tilde{A}}$ *is nondegenrate, and* $\deg F_{\tilde{A}} = \deg F_A$.

(c) Let $\tilde{A} \in B_F(A, \varepsilon)$ *be a random matrix with the uniform distribution on* $B_F(A, \varepsilon)$*, then* $\tilde{A}$ *is generic with probability one.*

Proof. (a) We have that

$$\min_{x \neq 0} \frac{\|F_A(x)\|_2}{\|x\|_2}$$

is zero if and only if $1 \in \mathrm{Spec}^{\mathrm{a}}(A)$ by Proposition 1.2.5. Hence, it is a positive number if $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$ and its inverse, the quantity $\kappa^{\mathrm{a}}(A)$, must be finite.

(b) By the inequalities between matrix norms, we can show that

$$\frac{1}{\kappa^{\mathrm{a}}(\tilde{A})} \geq \frac{1}{\kappa^{\mathrm{a}}(A)} - \|\tilde{A} - A\|\,.$$

Hence, if $\tilde{A} \in B(A, \varepsilon)$, with the given choice of ε, then no matrix in the segment $[A, \tilde{A}]$ can have 1 as an aligned value. Consequently, we have a path between A and $\tilde{A}$, given by $t \mapsto (1-t)A + t\tilde{A}$, such that no matrix in the path has 1 as an aligned value, and so $\deg F_{\tilde{A}} = \deg F_A$ by Proposition 1.3.4 (1).

(c) This is a direct consequence of Proposition 1.4.2. ☐

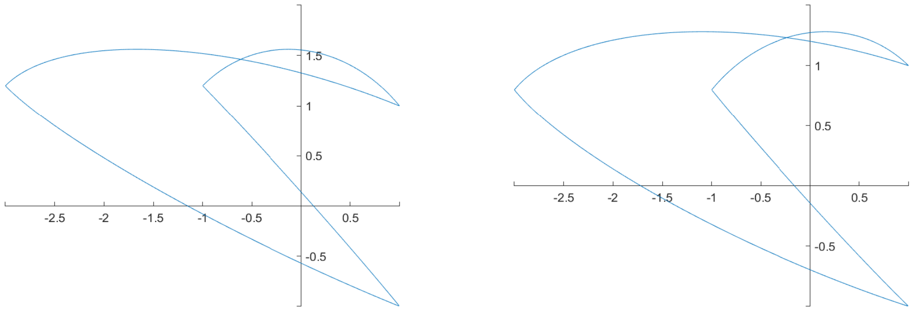


Figure 2: Image of the unit circle under F_{A_ε}. For $\varepsilon = 0.2$ it winds around the origin (left). For $\varepsilon = -0.2$ it does not (right).

We observe that the above proposition can only be applied when $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$. In that case, it allows us to produce a generic matrix $\tilde{A}$ such that $F_{\tilde{A}}$ has the same topological structure—degree—as F_A. Now, if $1 \in \mathrm{Spec}^{\mathrm{a}}(A)$, this perturbation trick will not produce $F_{\tilde{A}}$ with the same topological structure as the following examples show.

Example 1.4.6. *Let*

$$A := \begin{pmatrix} 2 & -1 \\ -1 & 0 \end{pmatrix} \tag{1.18}$$

for which $\mathrm{Spec}^{\mathrm{a}}(A) = \{1, \sqrt{2} - 1\}$. *Now consider the following perturbation*

$$A_\varepsilon := \begin{pmatrix} 2 & -1-\varepsilon \\ -1 & 0 \end{pmatrix}. \tag{1.19}$$

We can see that for $\varepsilon > 0$, $\deg F_{A_\varepsilon} = 1$, *since all aligned values are smaller than one; and that for* $\varepsilon < 0$, $\deg F_{A_\varepsilon} = 0$, *after a straightforward computation. Moreover, Figure 2 indicates—and it can be checked by computation—that*

$$F_{A_\varepsilon}(x) = \begin{pmatrix} r \\ 0 \end{pmatrix},$$

where $r > 0$, *does not have a solution for* $\varepsilon < 0$, *cf. Figure 2.*

Hence, when we perturb $A \in \mathrm{M}_n(\mathbb{R})$ *with* $1 \in \mathrm{Spec}^{\mathrm{a}}(A)$, *neither do we get consistent topological information about* $F_{\tilde{A}}$ *via the perturbed matrix* $\tilde{A}$, *nor do we obtain consistent information about the general solvability of the AVE* (1.2).

Example 1.4.7. *Consider a generic matrix* $A \in \mathrm{M}_n(\mathbb{R})$ *so that all aligned values have odd multiplicity. Then for every* $t > 0$, tA *is generic as long as* $1/t \notin \mathrm{Spec}^{\mathrm{a}}(A)$. *As* t *increases from zero to infinity, we have that* $\deg F_{tA}$ *alternates parity as* t *crosses the inverses of the aligned values of* A, *by Theorem 1.1.1. This shows again that perturbing matrices with 1 as an aligned value will not produce consistent topological information.*

1.5 Transfer of results to LCPs

Classically [CPS92, Chap. 1], the LCP is associated to the following piecewise linear function

$$\begin{aligned} G_M : \mathbb{R}^n &\to \mathbb{R}^n \\ z &\mapsto (M + \mathbb{I})z/2 + (M - \mathbb{I})|z|/2\,, \end{aligned} \tag{1.20}$$

where we have rewritten the expression in terms of absolute values. If G_M is surjective, then $\mathrm{LCP}(q, M)$ has a solution for any $q \in \mathbb{R}^n$. Therefore studying the degree of G_M is the *LCP*-equivalent of studying the degree of F_A in the context of AVEs.

The following proposition which is well-known in the literature (see [CPS92, Chap. 1], [MM06] and [Neu90, Chap. 6]) makes explicit how AVEs, LCPs and G_M relate.

Proposition 1.5.1. *Let $M \in \mathrm{M}_n(\mathbb{R})$ and $q \in \mathbb{R}^n$. The map $(v, w) \mapsto v - w$ is a bijective correspondence between the solution $(v, w) \in \mathbb{R}^n_{\geq 0} \times \mathbb{R}^n_{\geq 0}$ of* $\mathrm{LCP}(q, M)$ *and the solutions $x \in \mathbb{R}^n$ of $G_M(x) = q$. In particular, if $M + \mathbb{I}$ is invertible, then* $\mathrm{LCP}(q, M)$ *is equivalent to solve the AVE given by*

$$z - (M + \mathbb{I})^{-1}(M - \mathbb{I})|z| = 2(M + \mathbb{I})^{-1}q \qquad \square$$

In this case, the images of the spherical restrictions $\bar{F}_{(M+\mathbb{I})^{-1}(M-\mathbb{I})}$ and

$$\bar{G}_M \;\coloneqq\; \frac{G_M}{\|G_M\|_2}$$

are congruent, which shows that the degrees of both maps are identical up to a sign change according to the determinant sign of the transformation matrix $M + \mathbb{I}$. In particular, the parities of their degrees are the same.

To further analyze the degree of G_M, we introduce the LCP-variant of aligned trios of Definition 1.2.4 and of generic matrices of Definition 1.4.1—for which we can prove claims analogous to those of Section 1.4.

Definition 1.5.2. *Let $M \in \mathrm{M}_n(\mathbb{R})$. An* LCP-aligned trio *of M is a triplet $(\lambda, S, v) \in \mathbb{R}_{\geq 0} \times \mathcal{S} \times (\mathbb{S}^{n-1} \cap \mathbb{R}^n_{\geq 0})$ such that*

$$(M - \mathbb{I})v \;=\; \lambda(M + \mathbb{I})Sv\,.$$

Given such a trio, we call (S, v) an LCP-aligned vector *and λ an* LCP-aligned value *of M. The* LCP-aligned spectrum *of M, which we denote $\mathrm{Spec}^{\mathrm{a}}_{\mathrm{L}}(M)$, is the set of LCP-aligned values of M.*

LCP-aligned vectors are not eigenvectors of G_M and thus also not fixed points of $\bar{G}_M$. However, they and their polar opposites are exactly the pairs of antipodal points which are mapped to pairs of antipodal points by $\bar{G}_M$ (if the corresponding LCP-aligned value is smaller than 1) or to a single point (if the corresponding LCP-aligned value is larger than 1). This property of LCP-aligned vectors seems to be more crucial to the parity of the degree which mirrors the number of times that the sphere is folded onto itself by $\bar{G}_M$ than the property of being an eigenvector or a fixed point of the spherical map.

Definition 1.5.3. *An LCP-generic matrix $M \in \mathrm{M}_n(\mathbb{R})$ is a matrix such that a) $M + \mathbb{I}$ is invertible, and b) $(M + \mathbb{I})^{-1}(M - \mathbb{I})$ is generic.*

LCP-generic matrices are really generic due to Proposition 1.4.2 and the fact that $\{M \in \mathrm{M}_n(\mathbb{R}) \mid \det(M+\mathbb{I}) = 0\}$ is an algebraic hypersurface. Let $M \in \mathrm{M}_n(\mathbb{R})$ be a matrix so that G_M is nondegenerate. Analogously to Proposition 1.4.5, a random perturbation of M will be LCP-generic almost surely.

We can now state the main theorems (Theorems 1.2.2 and 1.1.4) and their corollaries in the context of LCPs. Recall that $\deg G_M$ is the degree of the spherical map $\bar{G}_M : x \mapsto G_M(x)/\|G_M(x)\|_2$.

Theorem 1.5.4. *Let $M \in \mathrm{M}_n(\mathbb{R})$ be LCP-generic such that $1 \notin \mathrm{Spec}^{\mathrm{a}}_{\mathrm{L}}(M)$. Then the degree of G_M is well-defined and it satisfies that*

$$\deg G_M \;\equiv\; 1 + \mathrm{c}^{\mathrm{a}}_{\mathrm{L}}(M) \mod 2$$

where $\mathrm{c}^{\mathrm{a}}_{\mathrm{L}}(M)$ is the number of LCP-aligned values greater than one, counted with multiplicity. Moreover, we have $\deg G_M = \mathrm{sign}(\det(M+\mathbb{I}))$ if $\mathrm{c}^{\mathrm{a}}_{\mathrm{L}}(M) = 0$ and $\deg G_M = 0$ if all LCP-aligned values of M are larger than 1.

Theorem 1.5.5. *Let $M \in \mathrm{M}_n(\mathbb{R})$ be LCP-generic such that $1 \notin \mathrm{Spec}^{\mathrm{a}}_{\mathrm{L}}(M)$ Then the degree of G_M is well-defined and it satisfies that*

$$\deg G_M \;=\; \mathrm{sign}(\det(M+\mathbb{I})) - \sum_{\substack{(\lambda,S,v)\ \text{LCP-aligned trio of } M \\ \lambda>1}} \mathrm{sign}\left(\chi'_{M+\mathbb{I},(M-\mathbb{I})S}(\lambda)\right).$$

where $\chi_{U,V} := \det(TU - V)$ is the generalized characteristic polynomial of $(U,V) \in \mathrm{M}_n(\mathbb{R})^2$.

Corollary 1.5.6. *Let $M \in \mathrm{M}_n(\mathbb{R})$ be an LCP-generic matrix. Then the number of LCP-aligned values of M, counted with multiplicity, is odd.* ☐

Corollary 1.5.7. *Let $M \in \mathrm{M}_n(\mathbb{R})$ be LCP-generic such that $1 \notin \mathrm{Spec}^{\mathrm{a}}_{\mathrm{L}}(M)$. If $\mathrm{c}^{\mathrm{a}}_{\mathrm{L}}(M)$ is even, then M is a Q-matrix, i.e., for all $q \in \mathbb{R}^n$, $\mathrm{LCP}(q,M)$ has a solution.*

To show these theorems and their corollaries, we only need to show how the notions in the LCP-setting crrespond to the notions in the AVE-setting. The following three propositions allow us to do these trasnlations. The first one gives how aligned trios correspond to LCP-aligned trios; the second one gives sufficient condition for the degree of G_M to be well-defined; and the third one relates the degree of G_M to that of F_A for an appropriate A.

Proposition 1.5.8. *Let $M \in \mathrm{M}_n(\mathbb{R})$ be such that $M+\mathbb{I}$ is invertible. Then $(\lambda, S, v) \in \mathbb{R}_{\geq 0} \times \mathcal{S} \times (\mathbb{S}^{n-1} \cap \mathbb{R}^n_{\geq 0})$ is an LCP-aligned trio of M if and only if it is an aligned trio of $(M+\mathbb{I})^{-1}(M-\mathbb{I})$.*

Proposition 1.5.9. *Let $M \in \mathrm{M}_n(\mathbb{R})$. Then $G_M(z) = 0$ has non-trivial solutions if and only if $1 \notin \mathrm{Spec}^{\mathrm{a}}_{\mathrm{L}}(M)$. In particular, if $1 \notin \mathrm{Spec}^{\mathrm{a}}_{\mathrm{L}}(M)$, the degree of G_M is well-defined, the set of regular values of G_M is dense and for all $y \in \mathrm{Reg}G_M$ we have*

$$\deg G_M = \sum_{x \in G_M^{-1}(y)} \mathrm{sign}(\det(\partial G_M(x))). \tag{1.21}$$

Proposition 1.5.10. *Let $M \in \mathrm{M}_n(\mathbb{R})$ be such that $1 \notin \mathrm{Spec}^{\mathrm{a}}_{\mathrm{L}}(M)$ and such that $M+\mathbb{I}$ is invertible. Then*

$$\deg G_M \;=\; \mathrm{sign}(\det(M+\mathbb{I})) \cdot \deg F_{(M+\mathbb{I})^{-1}(M-\mathbb{I})}.$$

Proof of Proposition 1.5.8. Note that, when $M+\mathbb{1}$ is invertible, (λ, S, v) is a LCP-aligned trio of M if and only if $\lambda S v = (M+\mathbb{1})^{-1}(M-\mathbb{1})v$. The latter is equivalent to $\lambda v = S(M+\mathbb{1})^{-1}(M-\mathbb{1})v$, which means exactly that (λ, S, v) is an aligned trio of $(M+\mathbb{1})^{-1}(M-\mathbb{1})$. ☐

Proof of Proposition 1.5.9. The first claim is proven as in Proposition 1.2.5. The second one follow from [CPS92, p. 509 ff], since G_M is a nondegenrate positively homogeneous map. ☐

Proof of Proposition 1.5.10. This follows from the multiplicative property of the degree and the fact that

$$L_M \circ \bar{F}_{(M+\mathbb{1})^{-1}(M-\mathbb{1})} = \bar{G}_M$$

where $L_M : x \mapsto (M+\mathbb{1})x/\|(M+\mathbb{1})x\|_2$. Recall that we have that $\deg L_M = \operatorname{sign}(\det(M+\mathbb{1}))$. ☐

We now give the proof of Theorems 1.5.4 and 1.5.5.

Proof of Theorem 1.5.4. By Propositions 1.5.8, 1.5.9 and 1.5.10, we have that

$$\deg G_M \;\equiv\; \deg F_{(M+\mathbb{1})^{-1}(M-\mathbb{1})} \mod 2$$

and

$$\mathrm{c}^{\mathrm{a}}_{\mathrm{L}}(M) \;\equiv\; \mathrm{c}^{\mathrm{a}}((M+\mathbb{1})^{-1}(M-\mathbb{1})) \mod 2.$$

Hence the theorem follows by Theorem 1.1.1. ☐

Proof of Theorem 1.5.5. By Propositions 1.5.8,

$$\deg G_M \;=\; \operatorname{sign}(\det(M+\mathbb{1}))\deg F_{(M+\mathbb{1})^{-1}(M-\mathbb{1})}.$$

Hence, by Theorem 1.1.4 and Proposition 1.5.10,

$$\deg G_M = \operatorname{sign}(\det(M+\mathbb{1})) - \sum_{\substack{(\lambda,S,v) \text{ LCP-aligned trio of } M \\ \lambda>1}} \operatorname{sign}\left(\det(M+\mathbb{1})\chi'_{(M+\mathbb{1})^{-1}(M-\mathbb{1})S}(\lambda)\right).$$

Now,

$$\det(M+\mathbb{1}))\chi'_{(M+\mathbb{1})^{-1}(M-\mathbb{1})S}(\lambda) = \chi'_{M+\mathbb{1},(M-\mathbb{1})S}(\lambda),$$

so we obtain the desired result. ☐

1.6 Proof of Theorems 1.1.1 and 1.1.4 and Corollaries 1.1.2 and 1.1.3

We first show how Theorem 1.1.1 follows from Theorem 1.1.4. Then we prove the corollaries 1.1.2 and 1.1.3. We finish giving the proof of Theorem 1.1.4.

1.6.1 Proof of Theorem 1.1.1

Note that the formula of Theorem 1.1.1 can be rewritten as

$$\deg F_A = 1 - \sum_{\substack{(\lambda,S,v) \text{ aligned trio of } A\\ \lambda>1}} \operatorname{sign}(\chi'_{SA}(\lambda)).$$

Now, since A is generic, we have that for each aligned trio (λ, S, A), λ is a simple eigenvalue of SA and so a simple root of χ_{SA}. Hence $\operatorname{sign}(\chi'_{SA}(\lambda))$ is either $+1$ or -1 for each summand in the sum. Moreover, for a specific aligned value λ, the summand $\operatorname{sign}(\chi'_{SA}(\lambda))$ appears as many times as the size of

$$\{(S,v) \mid S \in \mathcal{S},\ v \in \mathbb{S}^{n-1} \cap \mathbb{R}^n_{\geq 0}, (S,v) \text{ is the aligned vector of } A \text{ corresponding to } \lambda\}.$$

But this quantity, for generic A, is precisely the multiplicity of λ. Hence, we have that for all $\lambda > 1$,

$$\sum_{(\lambda,S,v) \text{ aligned trio of } A} \operatorname{sign}(\chi'_{SA}(\lambda))$$

is the multiplicity of λ mod 2. Hence

$$\sum_{\substack{(\lambda,S,v) \text{ aligned trio of } A\\ \lambda>1}} \operatorname{sign}(\chi'_{SA}(\lambda)) \equiv \mathrm{c^a}(A) \mod 2,$$

and the first part of the theorem follows.

The second part is just Proposition 1.3.4(2)-(3).

1.6.2 Proof of Corollary 1.1.2

Since A is generic, the multiplicity of 0 as an aligned value is given by the size of the set

$$\{(S,v) \mid S \in \mathcal{S},\ v \in \mathbb{S}^{n-1} \cap \mathbb{R}^n_{\geq 0}, (S,v) \text{ is the aligned vector of } A \text{ corresponding to } 0\}.$$

This set is invariant under the transformation $(S, v) \mapsto (-S, v)$. Hence, the multiplicity of 0 as an aligned value is always even.

By the proof of Lemma 1.6.1, we can consider perturbed matrix $\tilde{A}$ with the same number of positive aligned values as A, but such that 0 is not an aligned value of $\tilde{A}$. Since the multiplicity of 0 is even, this means that the parity of the number of aligned values of $\tilde{A}$ and A is the same. Therefore, without loss of generality, we can assume that A has only positive aligned values.

For $t > 0$, we have that

$$\mathrm{Spec^a}(tA) = t\,\mathrm{Spec^a}(A). \tag{1.22}$$

For some $t > 0$ sufficiently large, every aligned value of tA is larger than one, because, by assumption, A has only positive aligned values. Thus, by the second part of Theorem 1.1.1, $\deg F_{tA} = 0$, and so, by the first part of the same theorem, $\mathrm{c^a}(tA)$ is odd. Hence tA has an odd number of aligned values, as we wanted to show.

1.6.3 Proof of Corollary 1.1.3

If $\mathrm{c}^{\mathrm{a}}(A)$ is even, then, by Theorem 1.1.1, $\deg F_A$ is odd, and so, in particular, non-zero. Hence, by Proposition 1.3.2(4), $\bar{F}_A$ is surjective, and so is F_A.

1.6.4 Proof of Theorem 1.1.4

By perturbing A randomly, we can assume without loss of generality that all aligned values of A are simple, i.e., that every aligned value of A appears only in one aligned trio (λ, S, v) of A. The following lemma allows us to do this.

Lemma 1.6.1. *Let $A \in \mathrm{M}_n(\mathbb{R})$ be generic such that $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$. Then there is $\tilde{A}$ such that 1) $1 \notin \mathrm{Spec}^{\mathrm{a}}(A)$, 2) $\tilde{A}$ is generic, 3) F_A and $F_{\tilde{A}}$ have the same degree, 4) A and $\tilde{A}$ have the same aligned count, i.e., $\mathrm{c}^{\mathrm{a}}(A) = \mathrm{c}^{\mathrm{a}}(\tilde{A})$; and*

$$\sum_{\substack{(\tilde{\lambda},S,v) \text{ aligned trio of } \tilde{A} \\ \lambda>1}} \mathrm{sign}(\chi'_{S\tilde{A}}(\tilde{\lambda})) = \sum_{\substack{(\lambda,S,v) \text{ aligned trio of } A \\ \lambda>1}} \mathrm{sign}(\chi'_{SA}(\lambda));$$

and 5) for every aligned value $\tilde{\lambda}$ of $\tilde{A}$, there is a unique $S \in \mathcal{S}$ such that $\tilde{\lambda}$ is an eigenvalue of $S\tilde{A}$.

Proof. By Proposition 1.4.5, 1), 2), and 3) are guaranteed by taking $\tilde{A}$ in a sufficiently small neighborhood $B_F(A, \varepsilon)$ of A. Now, since A is generic, for each $S \in \mathcal{S}$, given an eigenvalue λ of SA, we proceed as follows:

(a) If $\lambda \notin [1, \infty) \subseteq \mathbb{C}$, then, by continuity of the eigenvalues, we have that for an arbitrarily small perturbation of A, $\tilde{A}$, the corresponding eigenvalue, $\tilde{\lambda}$, is still outside $[1, \infty)$.

(b) If $\lambda > 1$ is not an aligned value, then $(\lambda\mathbb{I} - SA)v$ does not vanish for $v \in \mathbb{S}^{n-1} \cap \mathbb{R}^n_{\geq 0}$. Therefore

$$\min_{v \in \mathbb{S}^{n-1} \cap \mathbb{R}^n_{\geq 0}} \|(\lambda\mathbb{I} - SA)v\|_2 > 0.$$

But this quantity is not only continuous in λ and A, but 1-Lipschitz in them. Hence, for an arbitrarily small perturbation $\tilde{A}$ of A, we can guarantee that the corresponding eigenvalue $\tilde{\lambda}$ does not become an aligned value of $\tilde{A}$.

(c) If $\lambda > 1$ is an aligned value, consider an aligned trio (λ, S, v) such that $\|v\|_2 = 1$. Since A is generic, λ is simple, and so we can apply the implicit function theorem to

$$\mathbb{R} \times \mathbb{S}^{n-1} \times \mathrm{M}_n(\mathbb{R}) \ni (\lambda, x, M) \mapsto (\det(\lambda\mathbb{I} - MS), (\lambda\mathbb{I} - MS)v)$$

at (λ, v, A). Hence, in a small neighborhood of A, we can write λ and v as smooth functions of A. Since v is strictly positive, this means that for an arbitrarily small perturbation $\tilde{A}$ the aligned value λ goes to an aligned value $\tilde{\lambda}$ which is still simple as an eigenvalue of $\tilde{A}S$. Moreover, if the perturbation is sufficiently small, $\tilde{\lambda}$ remains inside $(1, \infty)$, by continuity, and the signs of $\chi'_{SA}(\lambda)$ and of $\chi'_{S\tilde{A}}(\tilde{\lambda})$ coincide, by the continuity of $\chi'_{SM}(\mu)$ with respect to (M, μ).

Putting the above together, we have that taking $\tilde{A}$ in a sufficiently small neighborhood $B_F(A, \varepsilon)$ of A we can guarantee that it satisfies 1), 2), 3) and 4).

For 5), we only have to show that for almost all $\tilde{A} \in \mathrm{M}_n(\mathbb{R})$, the $S\tilde{A}$, with $S \in \mathcal{S}$, do not share any eigenvalue. Once this is done, we can guarantee, by the above, that we can choose a perturbation $\tilde{A}$ with the desired properties. Note that this is the only point where the perturbation is not arbitrary.

We will show that the set

$$M_S := \{X \in \mathrm{M}_n(\mathbb{R}) \mid X \text{ and } SX \text{ have an eigenvalue in common}\}$$

is a proper algebraic hypersurface. Then the set of X such that S_1X and S_2X share an eigenvalue for some $S_1, S_2 \in \mathcal{S}$ is given by

$$\bigcup_{S,T \in \mathcal{S}} SM_T.$$

Therefore it will be a proper algebraic hypersurface, since it is a finite union of proper algebraic hypersurfaces. Hence, we can choose $\tilde{A} \in B_F(A, \varepsilon)$ such that $\tilde{A}$ does not lie in it.

The determinant of the Sylvester matrix of the characteristic polynomials of X and SX is zero if and only if X and SX share an eigenvalue (see [CLO07, Ch. 3, Prop. 8]). Hence M_S is described by the zero set of a single polynomial. If it is not the full set $\mathrm{M}_n(\mathbb{R})$, then it is a proper algebraic hypersurface, as we wanted to show.

We show that M_S does not contain all matrices, by constructing a matrix not in it. Without loss of generality, we can assume that

$$S = \begin{pmatrix} \mathbb{1}_r & \\ & -\mathbb{1}_{n-r} \end{pmatrix}$$

with $r < n$. Consider the matrix

$$A = \begin{pmatrix} 0 & -\mathbb{1}_r & & & \\ 1 & 0 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & n-r-1 \end{pmatrix}$$

whose eigenvalues are $1, \mathrm{e}^{\frac{2\pi i}{r+1}}, \mathrm{e}^{\frac{4\pi i}{r+1}} \dots, \mathrm{e}^{\frac{2r\pi i}{r+1}}, 1, \dots, n-r-1$. Now, for S as defined above, we have that

$$SA = \begin{pmatrix} 0 & -\mathbb{1}_r & & & \\ -c & 0 & & & \\ & & -a_1 & & \\ & & & \ddots & \\ & & & & -a_{n-r-1} \end{pmatrix}.$$

Hence the eigenvalues of SA are $\mathrm{e}^{\frac{\pi i}{r+1}}, \mathrm{e}^{\frac{3\pi i}{r+1}}, \mathrm{e}^{\frac{5\pi i}{r+1}} \dots, \mathrm{e}^{\frac{(2r+1)\pi i}{r+1}}$, $-1, \dots, -(n-r-1)$, and so A and SA do not have common eigenvalues for a sufficiently general choice of c and the a_i. Hence $M_S \neq \mathrm{M}_n(\mathbb{R})$ as we wanted to show. ☐

We will be considering the map F_{tA} as t goes from arbitrarily small values to 1. By (1.22), the map F_{tA} is non-degenerate as long as $1/t \notin \mathrm{Spec}^{\mathrm{a}}(A)$. Let

$$1 < \lambda_1 < \cdots < \lambda_c$$

be the aligned values greater than one of A. Further, let $S_1, \ldots, S_c \in \mathcal{S}$ be the corresponding sign matrices of their unique aligned trios, and λ_0 be the largest aligned value smaller than 1 or zero if there are no such aligned values. By our initial assumption, they are not repeated.

When $t < 1/\lambda_c$, we have, by Proposition 1.3.4 (2), that F_{tA} has degree 1 and that $\mathrm{c}^{\mathrm{a}}(tA) = 0$. We aim to show that for $t \in (1/\lambda_{k+1}, 1/\lambda_k)$, we have

$$\deg F_{tA} = 1 - \sum_{i=k+1}^{c} \mathrm{sign}(\chi'_{AS_i}(\lambda_i)),$$

which gives the desired claim when $k = 0$. To do so, we only have to show that the degree of F_{tA} changes by $-\chi'_{AS_i}(\lambda_i)$ when t passes through $1/\lambda_i$. Note that, when t varies within $(1/\lambda_{k+1}, 1/\lambda_k)$, neither the aligned count nor the degree changes—the latter by Proposition 1.3.4(1). Hence, without loss of generality, it is enough to prove the following proposition.

Proposition 1.6.2. *Let* $A \in \mathrm{M}_n(\mathbb{R})$ *be generic and* $(1, \mathbb{1}, v)$ *one of its aligned trios. Assume that* $1 \in \mathrm{Spec}^{\mathrm{a}}(A)$ *is a simple aligned value such that for all* $S \in \mathcal{S} \setminus \{\mathbb{1}\}$, 1 *is not an eigenvalue of* SA. *Then there is an* $\varepsilon > 0$ *such that for all* $t, s \in (0, \varepsilon)$,

$$\deg F_{(1+t)A} = \deg F_{(1-s)A} - \mathrm{sign}(\chi'_A(1)).$$

Once this proposition is shown, we obtain the following proposition by applying the above one to AS/λ, where (λ, S, v) is the considered aligned trio.

Proposition 1.6.3. *Let* $A \in \mathrm{M}_n(\mathbb{R})$ *be generic and* (λ, S, v) *one of its aligned trios. Assume that* $\lambda \in \mathrm{Spec}^{\mathrm{a}}(A)$ *is a simple aligned value such that for all* $T \in\in \mathcal{S} \setminus \{S\}$, λ *is not an eigenvalue of* SA. *Then there is an* $\varepsilon > 0$ *such that for all* $t, s \in (0, \varepsilon)$,

$$\deg F_{(1/\lambda+t)A} = \deg F_{(1/\lambda-s)A} - \mathrm{sign}(\chi'_{AS}(\lambda)). \qquad \square$$

With this proposition, the desired claim follows, i.e., that F_{tA} changes the degree as wanted each time t passes through the inverse of an aligned value. Note that by our original perturbation, due to Lemma 1.6.1, the assumption is satisfied at each crossing.

Proof of Proposition 1.6.2

Since $\mathrm{Spec}^{\rho}(A)$ is discrete, there is some $\varepsilon_0 > 0$ such that $I_0 = (1 - \varepsilon_0, 1 + \varepsilon_0)$ does not contain other elements from $\mathrm{Spec}^{\rho}(A)$. For this interval, we have that for all $t \in I_0 \setminus \{1\}$, $\mathbb{1} - tA$ is invertible; and for all $t \in I_0$ and $S \in \mathcal{S} \setminus \{\mathbb{1}\}$, $\mathbb{1} - tAS$ is invertible. If $\mathbb{1} - tAS$ is not invertible, then $1/t \in \mathrm{Spec}^{\rho}(A)$ and so, by construction of I_0, $t = 1$ and, by assumption on the aligned value 1, $S = \mathbb{1}$.

Now, since 1 is an aligned value and A generic, let $v \in \mathbb{S}^{n-1} \cap \mathbb{R}^n_{>0}$ be the associated aligned vector with all positive entries. Now, we prove the following three lemmas in the context of Proposition 1.6.2.

Lemma 1.6.4. *Let $r > 0$. Then there exist $x \in B_2(-v, r)$ and $\delta > 0$ such that for all $t \in (1-\delta, 1+\delta) \setminus \{1\}$, x is a regular value of F_{tA} such that*

$$\sum_{\substack{z \in F_{tA}^{-1} \\ z \notin \mathbb{R}^n_{\geq 0}}} \operatorname{sign}(\det(\partial F_{tA}(z))) \tag{1.23}$$

does not depend on t.

Lemma 1.6.5. *There exist $r > 0$ and $\delta > 0$ such that for $t \in (1-\delta, 1)$,*

$$B(-v, r) \cap F_{tA}\left(\mathbb{R}^n_{\geq 0}\right) = \varnothing$$

and for $t \in (1, 1+\delta)$,

$$B(-v, r) \subset F_{tA}\left(\mathbb{R}^n_{\geq 0}\right).$$

Once these lemmas are proved, we only have to choose $r > 0$, $\delta > 0$ and $x \in B(-v, r)$ so that both Lemmas 1.6.5 and 1.6.4 apply. For this, we choose $r > 0$ and $\delta > 0$ as in Lemma 1.6.4, and then choose $x \in B(-v, r)$ and, if necessary, a smaller $\delta > 0$, following Lemma 1.6.5. In this way, for $s \in (1-\delta, 1)$, we obtain

$$\deg F_{sA} = \sum_{\substack{z \in F_{tA}^{-1} \\ z \notin \mathbb{R}^n_{\geq 0}}} \operatorname{sign}(\det(\partial F_{tA}(z)))$$

and for $t \in (1, 1+\delta)$,

$$\deg F_{tA} = \sum_{\substack{z \in F_{tA}^{-1} \\ z \notin \mathbb{R}^n_{\geq 0}}} \operatorname{sign}(\det(\partial F_{tA}(z))) + \operatorname{sign}(\det(\mathbb{1} - tA)),$$

since x has a preimage in the positive orthant when $t > 1$. Now, $\det(\mathbb{1} - tA) = \chi_{tA}(1)$. We have that $\chi_A(1) = 0$. Thus, for $t \in (1, 1+\delta)$ sufficiently small,

$$\operatorname{sign}(\det(\mathbb{1} - tA)) = \operatorname{sign}\left(\left.\frac{d}{dt}\right|_{t=1} \chi_{tA}(1)\right).$$

Now, $\chi_{tA}(1) = \sum_{k=0}^{n} c_k(A) t^{n-k}$, where $c_k(A)$ is the kth coefficient. Thus

$$\left.\frac{d}{dt}\right|_{t=1} \chi_{tA}(1) = \sum_{k=0}^{n} (n-k) c_k(A) = n \sum_{k=0}^{n-1} c_k(A) - \sum_{k=0}^{n-1} k c_k(A) = -\sum_{k=0}^{n} k c_k(A) = -\chi_A'(1),$$

where $\sum_{k=0}^{n-1} c_k(A) = -c_n(A)$ follows from $\chi_A(1) = 0$. Hence, the desired formula follows.

We finish the proof, by proving the lemmas above.

Proof of Lemma 1.6.4. Denote by

$$\mathcal{H} := \{y \in \mathbb{R}^n \mid \text{there is some } i \text{ such that } x_i = 0\} \tag{1.24}$$

the union of the coordinate hyperplanes, and fix $t \in I_0 \setminus \{1\}$. Then for all $x \in \mathbb{R}^n$, x is a regular point of F_{tA} if and only if $x \notin F_{tA}(\mathcal{H})$. If a preimage z of x does not lie on $\mathcal{H}$, then F_{tA} is differentiable at z. Moreover, at that point, the Jacobian will be of the form $\mathbb{1} - tAS$, for some $S \in \mathcal{S}$, and it will be invertible since $t \in I_0 \setminus \{1\}$.

Now, for all $t \in I_0$,

$$\tilde{\mathcal{H}}_t := \bigcup_{S \in \mathcal{S}} (\mathbb{1} - tAS)\mathcal{H} \supseteq F_{tA}(\mathcal{H})$$

is a hyperplane arrangement. Therefore we can choose a strictly negative $x \in B_2(-v, r)$ such that $x \notin \tilde{\mathcal{H}}_1$. If we show that $I_0 \ni t \mapsto \operatorname{dist}(x, \tilde{\mathcal{H}}_t)$ is a continuous map, then, since $\operatorname{dist}(x, \tilde{\mathcal{H}}_1) > 0$, we can choose $\delta > 0$ such that for all $t \in (1-\delta, 1+\delta)$, $\operatorname{dist}(x, \tilde{\mathcal{H}}_t) > 0$. Thus for all $t \in (1-\delta, 1+\delta)$, $x \notin F_{tA}(\mathcal{H})$ and, by the discussion in the above paragraph, x is a regular point of F_{tA}.

To show that $I_0 \ni t \mapsto \operatorname{dist}(x, \tilde{\mathcal{H}}_t)$ is continuous, we only have to observe that for all i, $S \in \mathcal{S}$ and $t \in I_0$,

$$(\mathbb{1} - tAS)\{x \mid x_i = 0\}$$

is a hyperplane, since $\{x \mid x_i\}$ does not contain any vector in the kernel of $\mathbb{1} - tAS$. Only $\mathbb{1} - A$ has a kernel, but it is a line spanned by the strictly positive vector v. Moreover, taking the generalized cross product of the $(\mathbb{1} - tAS)e_j$, $j \neq i$, and normalizing it, we can obtain a continuous map

$$I_0 \ni t \mapsto n(S, i, t) \in \mathbb{S}^{n-1}$$

that maps t to a normal vector of $(\mathbb{1} - tAS)\{x \mid x_i = 0\}$. Hence

$$\operatorname{dist}(x, \tilde{\mathcal{H}}_t) = \min \left\{ |n(S, i, t)^t x| \mid i \in \{1, \ldots, n\},\, S \in \mathcal{S} \right\}$$

is a continuous function as we wanted to show.

Now, for all $t \in (1-\delta, 1+\delta)$ and all $S \in \mathcal{S} \setminus \{1\}$, we have that

$$(\mathbb{1} - tAS)^{-1} x \notin \mathcal{H}.$$

Note that, by choosing $\delta > 0$ small enough if necessary, we can guarantee $t \in I_0$ and that $\mathbb{1} - tAS$ is invertible. Hence we have that the signs of $(\mathbb{1} - tAS)^{-1} x$ are constant. This means that, independently of $t \in (1-\delta, 1+\delta)$,

$$F_{tA}^{-1}(x) \cap S\mathbb{R}^n_{>0}$$

is either empty or has size one. Now, for $z \in S\mathbb{R}^n_{>0}$, $\operatorname{sign}(\det(\delta F_{tA}(z))) = \operatorname{sign}(\det(\mathbb{1} - tAS))$ is constant. Hence the sum (1.23) remains constant as desired. ☐

For the proof of Lemma 1.6.5, we will need the following auxiliary lemma.

Lemma 1.6.6. *Let $A \in \mathrm{M}_n(\mathbb{R})$. If λ is a simple eigenvalue of A, then*

$$\ker(\lambda\mathbb{1} - A) \cap \operatorname{im}(\lambda\mathbb{1} - A) = 0.$$

Proof of Lemma 1.6.6. If $v \in \ker(\lambda\mathbb{1} - A) \cap \operatorname{im}(\lambda\mathbb{1} - A)$ is non-zero, then $v \in \ker(\lambda\mathbb{1} - A)^2$, but $v \notin \ker(\lambda\mathbb{1} - A)$. Thus v is a nontrivial generalized eigenvector of rank 2 of A corresponding to λ. However, λ is a simple eigenvalue of A, so this is impossible. ☐

Proof of Lemma 1.6.5. By Lemma 1.6.6, we have that

$$-v \notin (\mathbb{I} - A)\mathcal{H},$$

where $\mathcal{H}$ is the union of the coordinate hyperplanes, as in the proof of Lemma 1.6.4. Arguing as in that proof, we have that $I_0 \ni t \mapsto \operatorname{dist}(-v, (\mathbb{I} - tA)\mathcal{H})$ is continuous, and so, since $\operatorname{dist}(-v, (\mathbb{I} - A)\mathcal{H})$ is positive, we have that there is $\delta > 0$ and $r > 0$ such that for all $t \in (1 - \delta, 1 + \delta) \subseteq I_0$,

$$\operatorname{dist}(-v, (\mathbb{I} - tA)\mathcal{H}) > r.$$

We show now that these are the desired r and δ.

Fix $t \in (1 - \delta, 1)$. $F_{tA}(\mathbb{R}^n_{\geq 0})$ is a closed pointed cone, since it is image of a closed pointed cone under the invertible linear map $\mathbb{I} - tA$. This cone contains v, so it does not contain $-v$. Now,

$$\operatorname{dist}(-v, F_{tA}(\mathbb{R}^n_{\geq 0})) = \operatorname{dist}(-v, \operatorname{bd} F_{tA}(\mathbb{R}^n_{\geq 0})) \geq \operatorname{dist}(-v, (\mathbb{I} - A)\mathcal{H}),$$

since the nearest point in $F_{tA}(\mathbb{R}^n_{\geq 0})$ to $-v$ lies in the boundary, $\operatorname{bd} F_{tA}(\mathbb{R}^n_{\geq 0})$ of $F_{tA}(\mathbb{R}^n_{\geq 0})$, which is contained inside $(\mathbb{I} - A)\mathcal{H}$. Hence, by the first paragraph in the proof, we have $\operatorname{dist}(-v, F_{tA}(\mathbb{R}^n_{\geq 0})) > r$, and so $B(-v, r) \cap F_{tA}(\mathbb{R}^n_{\geq 0})) = \varnothing$, as desired.

Fix $t \in (1, 1 + \delta)$. $F_{tA}(\mathbb{R}^n_{>0})$ is a full-dimensional cone, since it is the image of a full-dimensional cone under the invertible linear map $\mathbb{I} - tA$. Now, we have that

$$\operatorname{dist}(-v, \operatorname{bd} F_{tA}(\mathbb{R}^n_{\geq 0})) \geq \operatorname{dist}(-v, (\mathbb{I} - A)\mathcal{H}).$$

Therefore, by the first paragraph in the proof, $\operatorname{dist}(-v, \operatorname{bd} F_{tA}(\mathbb{R}^n_{\geq 0})) > r$, and so we obtain $B(-v, r) \subseteq F_{tA}(\mathbb{R}^n_{\geq 0}))$, as desired. ☐

Chapter 2

Calibration of internal combustion engines

This chapter is taken from the article “Semi-automatically optimized calibration of internal combustion engines” [BJP$^+$20] by Timo Burggraf, Michael Joswig, Marc E. Pfetsch, Manuel Radons, and Stefan Ulbrich which has been published in Optimization and Engineering (2020).

2.1 Introduction

Due to the wish to save fossil fuels, stringent maximal emission limits and challenging customer preferences, modern internal combustion engines (ICEs) become more and more complex. Indeed, research towards the improved construction of combustion engines is highly relevant for reaching the CO_2 emission targets set by the European Union, cf. [M$^+$16].

This engine development results in an increasing number of *actuators* and *sensors*. There are currently in the order of ten different actuators and sensors each. Examples for actuators include the amount of injected fuel, exhaust recirculation control, air valve angle, etc. Sensors measure, e.g., the temperature, maximal point of the cylinder, torque, exhaust emission, etc. The actuators allow to produce a certain torque and revolution frequency, which describe the two main requirements on the engine in usage. However, several different settings of the actuators can result in the same torque/revolution frequency combination. Moreover, their dependence is involved and not known exactly a priori, i.e., it has to be measured and approximated. In this paper we deal with the *optimal engine calibration problem*, i.e., to efficiently approximate this dependence by few measurements and to choose optimal actuators settings.

The engine calibration problem consists of determining a so-called *engine manifold*, which determines for each torque/revolution frequency combination a corresponding setting for actuators. This manifold is usually discretized and the resulting *solution map* is hard-coded into the engine control unit (ECU). The settings on the engine manifold have to be chosen in such a way that they are consistent across various torque/frequency combinations, i.e., the engine manifold should be continuous. Moreover, the resulting settings need to obey several restrictions in order to avoid damage of the engine as well as bounds on the emissions produced. The emissions are measured with respect to so-called *driving cycles*. These are certain prescribed changes in the torque/frequency settings over time which are supposed to resemble usage in practice. These driving cycles are applied to the engine on a measurement bench and the resulting emissions have to be bounded.

The calibration process described above involves two main steps. First, the dependence between the actuator settings and the output has to be measured. A naive enumeration of a uniform grid of possible actuator settings and interpolation would require exponential time in the number of actuators. Therefore, one needs to design a process to perform the measurements in relevant areas in order to speed up the measurement process. Additionally, the actuator settings are continuously changed without waiting until a steady-state has been reached. This allows to save time, but also introduces measurement errors like hysteresis effects. Second, based on the so-obtained information about the engine behavior, one should optimally choose the resulting engine manifold and solution map. More precisely, one should select actuator settings for the final solution map that obey the above-mentioned restrictions and yield an approximation to an optimal solution map which is sufficiently close in the sense that all prescribed emission targets are met.

In this article we propose a new way to solve the engine calibration problem, which consists of the following contributions:

- *Adaptive meshing:* The density of measurements is adapted within areas where the measured function is sensitive with respect to its inputs, while keeping the density of measurements coarse where it is not. This leads to a more accurate representation of the engine manifold than with a uniform grid approach with a fraction of the measurements. This first part is based on the well-established `LOLIMOT` (local linear model tree) [SHI00] partitioning scheme of the space of actuator settings, which we extend by more involved measurement-routing and grid-refinement schemes.
- *Data cleaning:* Before optimization, the measured data are cleaned by filtering out redundancies and noise.
- *Discretization and Optimization:* We discretize the space of measurands in a fashion that fits the format of lookup tables as they are stored in the engine control unit. Using discrete optimization techniques, we select among the measured (and cleaned) data such actuator settings that minimize fuel consumption of the engine while its pollutant emissions conform to current regulations. The selected settings are drivable in the sense that the actuator's variation speed is bounded in order to prevent engine damage.

We would like to stress that the algorithm described in this article is a *tool for* the engineer, not a replacement. While it runs automatically once its parameters have been set, the setting of these parameters, e.g., the determination of the subset of actuators which is to be varied in a given situation, requires extensive knowledge and experience. In this sense it is a semi-automatic process.

This article is structured as follows. In Section 2.2 we briefly review relevant literature. A high-level description of the mathematical problem is given in Section 2.3. From this, we arrive at the corresponding steps of our process. Section 2.4 provides the details of our method. In Section 2.5 we present a practical case-study, and Section 2.6 contains the experimental results. We close with a few remarks.

2.2 State of the Art

In this section we describe the state of the art of calibration methods for the optimization of ICEs. Many commercial and research products exist for measurement and calibration. In the following we give a brief overview of the main approaches. Any calibration process comprises the following two major components:

(A) Obtain a good approximation of the engine behavior, mathematically described in terms of some function.

(B) Use whatever (approximate) knowledge of that function to produce a set of lookup tables for the ECU.

The lookup tables may be optimized for various goals, e.g., dynamic driving behavior or maximal performance. In the present work we are set on optimizing the fuel consumption while conforming to a set of emission constraints.

2.2.1 A Naïve Measurement-Based Approach

A basic idea is to measure all actuator settings on a sufficiently fine uniform grid. For modern engines this approach is infeasible for two reasons. First, the size of the grid increases exponentially with the number of actuators. This would increase the number of measurements —and thus also the measurement time— beyond any feasible bounds. Second, even if such a comprehensive measurement were possible, it is computionally infeasible to optimize over such large input. So neither Step (A) nor Step (B) can be realized in this way.

2.2.2 The Model-Based Approach

For Step (A) in a model-based calibration the measurements are used to fit a given model to the engine behavior. Once this fitting process is completed, in Step (B) lookup tables for the ECU which are optimal with respect to the so-obtained model and a given set of objectives are computed via standard techniques from nonlinear optimization such as steepest descent methods, cf. [Ise14, p. 542]. A software package which does so in an automated fashion is the *Model-Based Calibration Toolbox* for `MATLAB` [Mat18]. Leading model-based approaches employ physical models, the training of neural networks via the measurements, and statistical machine learning. An implementation that combines the first two kinds is, e.g., the software package `mbminimize` [KPF$^+$03]. Statistical machine learning is used, e.g., in `ASCMO` [KKL10].

Physical models are usually described in terms of smooth functions. Similarly, most neural networks also model smooth functions. For instance, perceptrons are commonly composed of sigmoid functions and thus smooth, cf. [Ise03, pp. 103–111]. This restriction to mathematical modeling via smooth functions inherently imposes severe limitations to any model-based calibration of ICEs. The reason is that the functions necessary to describe an ICE well enough exhibit strong nonlinearities and may even be nondifferentiable. In the former case, the approximation error can be bounded globally, but locally the approximating functions are likely to exhibit strong oscillations that do not describe the underlying engine behavior very well; cf. [Ise03, p. 105]. At a nondifferentiability of a continuous function the role of the local linear approximation, which is the derivative, is taken over by local piecewise linear approximations [Sch12, p. 67ff]. In this case any sufficiently good approximation usually requires an exponential number of local models, one for each linear piece. As a consequence, it is common that either Step (A) does not describe the ICE well enough or both Step (A) and Step (B) are infeasible due to a combinatorial explosion, just like in the naïve approach.

2.2.3 LOLIMOT

Local linear model trees (`LOLIMOT`) are arguably the most performant model-based calibration method to date; cf. [Ise14, p. 93], as well as [MVTI03]. `LOLIMOT` partitions the space of actuator settings as a dissection into cubical cells. In each cubical cell, the engine behavior is modeled by a Gaussian function on the basis of a "central composite measurement point pattern"; cf. [Ise14, Fig. 3.4.12]. These local models are then stitched together into a global one. If a local model fails to produce a sufficiently close approximation, e.g., due to a strongly nonlinear behavior of the engine, the corresponding cubical cell is split along one of its axes and the engine behavior is modeled via a central composite design on each sub-cell. This process is iterated until a sufficient model quality is reached globally.

The key contribution of `LOLIMOT` is to introduce adaptive meshing to the engine calibration process. One drawback of the method is that only one output value is modeled at a time. The engine model is then fused together from the component models for the individual measurands; cf. [Ise14, p. 93]. This approach may cause both holes in the modeled behavior as well as local redundancies of data. Further, the model produced by `LOLIMOT` is a special type of radial basis function network; cf. [Ise10, p. 143]. These types of neural nets require a locally homogeneous covering of the actuator space by measurements; cf. [Ise03, p. 111]. This then necessitates the aforementioned fixed measurement patterns within each cell. But any such fixed pattern is again, like in the previous approaches, subject to a combinatorial explosion in high dimensions. The adaptive meshing with fixed local pattern is better than the uniform grid used in the naïve approach, but only by a constant factor. This is the second drawback.

Our new method for engine calibration refines `LOLIMOT` by simultaneously considering all measurands. This employs a more involved grid refinement scheme and randomized measurement routing; cf. Section 2.4.2. In this way we can overcome conceptual limitations of `LOLIMOT`.

2.2.4 Data Boundaries

Throughout its operation the measurand values of an ICE have to stay within certain boundaries. Some of these boundaries are set in place to avoid destruction, e.g., for cylinder pressures or critical device temperatures. Others are induced, e.g., by emission and noise regulations. In practice they are obtained automatically via established software packages such as `Cameo` [GPFL01], `TopExpert` [FEZ04], or the `MATLAB` toolbox `LOLIMOT` (local linear model tree) [SHI00]. Throughout we will assume that all such boundary informations are already given.

2.3 Mathematical Problem Description

The calibration of an internal combustion engine is the procedure to derive an engine manifold, which is optimal with respect to some predefined objective. To reduce the complexity of the model and to enhance practical implementation, these manifolds are discretized to obtain solution maps (see Section 2.3.2 below). The calibration problem posed informally in [MST18, p. 250] asks for an engine manifold that minimizes fuel consumption, while conforming to a number of emission constraints. In this section we will present two mathematical formalizations of the latter, one idealized continuous version and its discretization whose output fits the format of the lookup tables for the ECU.

In our setting, knowledge of the engine behavior with respect to variations of its actuator settings is obtained by means of physical experiments on a test bench. In addition to the revolution frequency, typical actuators include the injected fuel quantity, the injection angle, or the valve pressure. The generated torque of the engine is a measurand.

Technically there is a relevant distinction between direct and controlled actuators. Direct actuators such as injected fuel quantity, injection angle, or valve pressure can be set directly on the engine, while controlled actuators are set indirectly. For example, the revolution frequency is a controlled actuator that is regulated via a brake on the engine shaft. However, in our mathematical model this distinction is not relevant.

The aforementioned side constraints include limits on pollutant emission as well as physical requirements such as engine temperature limits. They necessitate that not only torque, but several other output values of the engine are measured as well. A realistic engine model features $m \geq 8$ actuators and $n \geq 14$ measurands. In our mathematical model we represent the relation between the setting of m actuators and n sensor values by a function

$$F\colon \mathbb{R}^m \to \mathbb{R}^n \, .$$

Throughout we make the fundamental assumption that F is continuous, but not necessarily differentiable everywhere. Further, we will assume the actuator values to be restricted to a box $U_{\mathrm{ad}} \subset \mathbb{R}^m$, which we call the *admissible domain.* It was already noted in Section 2.2.4 that throughout this work we assume U_{ad} to be already given. The non-critical sensor values define another box $Y_{\mathrm{ad}} \subset \mathbb{R}^n$, the *admissible range*; cf. Section 2.2.4. The exact definitions of U_{ad} and Y_{ad} will be stated in the subsequent Section 2.4.1. The *feasible space* of F is the set

$$F_{\mathrm{fsb}} \coloneqq \{(u, y) \in U_{\mathrm{ad}} \times Y_{\mathrm{ad}} \mid y = F(u)\} \;\subseteq\; \mathbb{R}^m \times \mathbb{R}^n \, .$$

The actuators correspond to the coordinates of u; examples are the revolution frequency and the amount of fuel injected. Typical sensor values, i.e., coordinates of $F(u)$, include the torque and the emission of carbon monoxide.

We denote by “freq” the index of the actuator for revolution frequency and by “torq” the index of the torque sensor values. Then we call

$$\mathrm{OP} := \{(u_{\mathrm{freq}}, y_{\mathrm{torq}}) \mid (u, y) \in F_{\mathrm{fsb}}\}$$

the *operation field* of F. The operation field represents the behavior of the engine with respect to revolution frequency and torque. While our methods are more general, we focus on this particular pair of actuator and sensor values in our analysis.

2.3.1 Continuous Optimization Problem

It is important to understand that the actual optimization problem is inherently discrete, since the desired output is a solution map for the ECU. These lookup tables of actuator values for given frequency and torque combinations have a given finite length. For the sake of a concise exposition, however, we now describe an idealized continuous optimization problem which has the actual optimization problem that we want to solve as a natural discretization. While similar optimization problems must be behind all known approaches to ICE calibration, we are not aware of any complete description in the available literature. Our model below is intended to fill this gap.

Optimization Space and Drivability

The feasible region of our continuous optimization problem is given by all engine manifolds. Each such manifold is given as the image of a map $M\colon \mathrm{OP} \to \mathbb{R}^m$ that assigns actuator settings to a given frequency and torque value pair in the operation field, i.e.,

$$[F(M(u_{\mathrm{freq}}, y_{\mathrm{torq}}))]_{\mathrm{torq}} \;=\; y_{\mathrm{torq}} \quad \text{for all } (u_{\mathrm{freq}}, y_{\mathrm{torq}}) \in \mathrm{OP} \;\; . \tag{2.1}$$

A basic requirement is that these maps are continuous.

Moreover, there are vital additional conditions to consider. Any solution to the engine calibration problem must be drivable in the sense of [MST18, p. 258]: Varying actuators too fast might damage the engine. Therefore in the (continuous) final solution map the variation speed of every actuator is bounded by constants Δ_a for $a \in \{1, \ldots, m\}$. For a given map M and actuator a, the corresponding *drivability constraint* is that for all $(u_{\mathrm{freq}}, y_{\mathrm{torq}})$, $(u'_{\mathrm{freq}}, y'_{\mathrm{torq}}) \in \mathrm{OP}$ with $u := M(u_{\mathrm{freq}}, y_{\mathrm{torq}})$ and $u' := M(u'_{\mathrm{freq}}, y'_{\mathrm{torq}})$ the following has to hold

$$|u_a - u'_a| \;=\; |[M(u_{\mathrm{freq}}, y_{\mathrm{torq}})]_a - [M(u'_{\mathrm{freq}}, y'_{\mathrm{torq}})]_a| \;\leq\; \Delta_a \cdot \|(u_{\mathrm{freq}}, y_{\mathrm{torq}}) - (u'_{\mathrm{freq}}, y'_{\mathrm{torq}})\| \, , \tag{2.2}$$

where $\|\cdot\|$ is some norm. As a consequence, the map $M_a : \mathrm{OP} \to \mathbb{R}$ is Lipschitz continuous with constant Δ_a. We define Ω as the set of all maps M which are feasible in the sense that they obey the drivability constraint with respect to each actuator, more precisely,

$$\Omega \;:=\; \big\{M\colon\, \mathrm{OP} \to \mathbb{R}^m\colon\; M \text{ satisfies (2.1) and moreover (2.2) for all } a \in \{1, \ldots, m\}\big\}.$$

Driving Cycle and Emission Constraints

In accordance with current government regulations and common test cycles, the engine behavior is optimized with respect to pre-defined scenarios, known as driving cycles. In

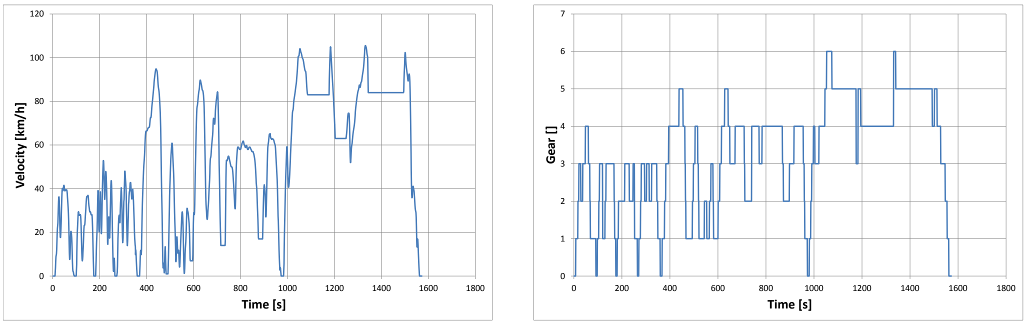


Figure 3: Operational profile of RANDOM driving cycle.

our continuous model a *driving cycle* is a time-parametrized curve

$$\gamma_{\mathrm{dc}} \colon [0,1] \to \mathrm{OP}\,,$$

whose purpose is to simulate the phases of acceleration and constant speed of real-world driving patterns. Driving cycles are given indirectly in the form of operational profiles which map a time to a gear/velocity combination; cf. Figure 3. Taking into account the weight of the car, its drag coefficient, specific tire friction, etc., every such gear/velocity combination can be mapped to a revolution frequency/torque combination in the operation field.

For the sake of the simplicity of exposition, here we will assume that the calibration is performed with respect to a fixed driving cycle. Yet it is also natural to take the combination of several driving cycles into account, and our approach also covers this slightly more general situation.

An important constraint prescribed by regulations is to bound the resulting emissions along the driving cycle. Emission pollutants include carbon monoxide (`CO`), hydrocarbons (`HC`), nitrogen oxides ($\mathtt{NO_x}$) as well as particulate matter (`PM`) and number (`PN`). We denote by E the index set that corresponds to pollutant emissions and by $\mathfrak{e}_p$ the emission limit for pollutant p over the driving cycle.

Consider an engine manifold map $M \in \Omega$ and a pollutant $p \in E$. The frequency and torque values along γ_{dc} produce actuator settings $M(\gamma_{\mathrm{dc}}(t))$, which result in output values $F(M(\gamma_{\mathrm{dc}}(t)))$, including the pollutant p. The integral of these values must satisfy the *emission constraint* for all $p \in E$:

$$\int_0^1 [F(M(\gamma_{\mathrm{dc}}(t)))]_p \cdot \|\dot{\gamma}_{\mathrm{dc}}(t)\|\,\mathrm{d}t \;\leq\; \mathfrak{e}_p\,. \tag{2.3}$$

The factor $\|\dot{\gamma}_{\mathrm{dc}}(t)\|$ accounts for the fact that γ_{dc} is not necessarily normalized, i.e., the speed of acceleration, deceleration, etc., varies. Table 2.1 shows the diesel engine emission constraints for EURO norms 3-6c (E3-E6c).

The optimization formulation

Let “fuel” be the index of the actuator for the injected fuel quantity. It is possible that one wants to optimize fuel consumption over a different curve than the driving cycle γ_{dc}.

Table 2.1: Selection of EURO emission constraints for passenger cars with compression ignition engine.

Em.	Unit	E3 (2001)	E4 (2006)	E5a (2011)	E6b (2015)	E6c (2018)
CO	mg / km	2300	1000	1000	1000	1000
HC	mg / km	200	100	100	100	100
$\mathrm{NO_x}$	mg / km	150	80	60	60	60
PM	mg / km	–	–	5	4.5	4.5
PN	1 / km	–	–	–	$6 \cdot 10^{12}$	$6 \cdot 10^{11}$

We thus define a second curve in the operation field

$$\varphi\colon [0,1] \to \mathrm{OP}\ .$$

Using the terminology developed so far, the continuous optimization problem is

$$\begin{aligned} \text{minimize}_{M\in\Omega} \quad & \int_0^1 [M(\varphi(t))]_{\text{fuel}} \cdot \|\dot{\varphi}(t)\|\,\mathrm{dt} \\ \text{subject to} \quad & \int_0^1 [F(M(\gamma_{\text{dc}}(t)))]_p \cdot \|\dot{\gamma}_{\text{dc}}(t)\|\,\mathrm{d}t \ \le\ \mathfrak{e}_p \quad \text{for all } p \in E\,. \end{aligned} \tag{2.4}$$

That is, the optimization goal is to minimize the total consumed fuel with respect to the time parametrized curve φ in OP, while for all pollutants p the integral emission along γ_{dc} is bounded by the prescribed constant $\mathfrak{e}_p$. Implicitly, problem (2.4) is subject to the driviability constraint, due to the fact that all manifold maps $M \in \Omega$ must conform to inequality (2.2). In the discrete formulation of the engine calibration problem this dependence will be made explicit. We further remark that the curves φ and γ_{dc} may coincide, but they do not have to. More elaborate curves φ may be useful, e.g., for controlling the fuel consumption also outside the driving cycle, while ignoring the fringes of the operation field.

2.3.2 Discrete Optimization Problem

To obtain a finite-dimensional problem, the model is discretized to yield *(characteristic) engine maps.* We consider combinations of k revolution frequencies and k torque demands and subdivide the operation field OP of F into k^2 congruent rectangles OP_{ft}, where f and $t \in [k] := \{1, \dots, k\}$ denote the rectangle's frequency and torque coordinate, respectively. This corresponds to the technical requirement that engine maps are given as $k{\times}k$-matrices, which for each combination yield the corresponding value of a particular actuator. They are stored permanently in the engine control unit. A *solution map* consists of a complete set of engine maps, one for each actuator. In this way a solution map yields a discretization of the continuous solution map M described in Section 2.3.1.

Then, in our terminology, the solution map takes as input a frequency and torque pair $(f,t) \in [k] \times [k]$ and yields as output an admissible actuator setting. The latter is a point $u \in U_{\text{ad}}$ with $(u, F(u)) \in F_{\text{fsb}}$ such that the pair $(u_{\text{freq}}, y_{\text{torq}})$ lies in the rectangle of the discretized operation field corresponding to the input coordinates (f,t).

Our goal is to obtain a solution map which is optimal with respect to a given objective, while conforming to several constraints. Since typical constraints are continuous but nonlinear and the solution map itself is discrete, this optimization problem belongs to the wide class of mixed-integer/discrete nonlinear optimization problems, which are often difficult to handle.

Splitting the engine calibration into Steps (A) (data acquisition) and (B) (computation of a solution map), cf. Section 2.2, our goal can be formulated as follows. In Step (A) obtain, via actual measurements of the engine, a finite set

$$\widehat{F} \coloneqq \{(u^1, y^1), (u^2, y^2), \ldots, (u^\delta, y^\delta)\}$$

of δ points in F_{fsb}, i.e., a set of actuator settings $u^q \in U_{\mathrm{ad}}$ such that $y^q = F(u^q) \in Y_{\mathrm{ad}}$, which is a sufficiently good representation of F. Here "sufficiently good" means that we can, in Step (B), extract from $\widehat{F}$ a solution map that conforms to discretized versions of the drivability and emission constraints presented in Section 2.3.1. In this sense, a solution map is a sufficiently good discrete approximation to an optimal engine map M of the continuous optimization problem (2.4).

Since $\widehat{F}$ is obtained via actual measurements, it is called the *data set*. The elements

$$d^q \coloneqq (u^q, y^q)$$

of $\widehat{F}$ are called *data points*. The final selection of the data points is the result of a cycle of measurements and optimization steps, which are discussed in the sections below. We will assume that none of the points in the data set lies on the boundary of any rectangle OP_{ft}. In this case, the discretization of the operation field into k^2 rectangles partitions $\widehat{F}$ into sets

$$S_{ft} \coloneqq \left\{(u, y) \in \widehat{F} \;\middle|\; (u_{\mathrm{freq}}, y_{\mathrm{torq}}) \in \mathrm{OP}_{ft}\right\}.$$

We call each set S_{ft} a *stack*, and the entire partition

$$k\,\mathrm{OP} \coloneqq \{S_{ft} \mid f, t \in [k]\}$$

is the *k-operation field* of F with respect to $\widehat{F}$. In the solution map each entire rectangle OP_{ft} will be represented by a single measurement $d_{ft} \in S_{ft}$. Hence, the discrete analogue to Ω, the set of all manifold maps, is the set Ω_k of all maps $M_k \colon [k] \times [k] \to \mathbb{R}^m \times \mathbb{R}^n$, such that $M_k(f, t) \in S_{ft}$, i.e., select exactly one data point from each stack, and satisfy a discrete analogue of the drivability constraint (2.2), see (2.5) below. Thus,

$$\Omega_k \;\coloneqq\; \left\{M_k \colon [k] \times [k] \to \mathbb{R}^m \times \mathbb{R}^n \colon\; M_k(f, t) \in S_{ft}\ \forall f, t \in [k] \text{ and } M_k \text{ satisfies (2.5)}\right\}.$$

The elements of Ω_k are called solution maps. Each solution map $M_k \in \Omega_k$ has precisely k^2 values $M_k(f, t) \in S_{ft}$, one per stack. A solution map is uniquely determined by the data points $d_{ft} = M_k(f, t)$, $f, t \in [k]$, and we call d_{ft} the *representatives* of the k^2 stacks S_{ft} corresponding to M_k. The final calibration solution is a solution map $\mathrm{SOL} \in \Omega_k$ which is optimal with respect to the given objectives, i.e., minimization of fuel consumption, subject to emission regulations and drivability. The final solution will be picked by solving the optimization problem (2.8) below.

Discrete drivability constraint

Due to the uniform discretization of the k-operation field, *a discrete drivability constraint* merely has to bound the difference between actuator settings of representatives of neighboring rectangles OP_{ft}. That is, with $d_{ft} = M_k(f,t)$, the discrete analogue of (2.2) is

$$|[d_{ft}]_a - [d_{gs}]_a| \;=\; |[M_k(f,t)]_a - [M_k(f,t)]_a| \;\leq\; \Delta_a \qquad \text{for all } a \in [m]\,, \tag{2.5}$$

and all tuples $(f,t),(g,s) \in [k] \times [k]$, where either $g = f \pm 1$ or $s = t \pm 1$.

Discrete emission constraint

To discretize the emission constraint (2.3), we replace an engine map $M \in \Omega$ by its discrete counterpart $M_k \in \Omega_k$ and obtain instead of the curve integral along the curve γ_{dc} a weigthed sum. In fact, to each rectangle OP_{ft} we associate a weight ω_{ft}. This weight is set to zero if the intersection of OP_{ft} with the image of the curve γ_{dc} is empty. Otherwise ω_{ft} is a positive value that reflects the resistance time, i.e., the duration of the curve γ_{dc} staying in the rectangle OP_{ft}. The function

$$\mathrm{DC}\colon [k] \times [k] \to \mathbb{R}_{\geq 0},\; (f,t) \mapsto \omega_{ft} \tag{2.6}$$

serves as a discrete analogue of the continuous driving cycle γ_{dc}. Note that the map DC only records which parts of the operation field are met by γ_{dc} for which duration, but it ignores the order in which this happens.

The *discrete emission constraint* is now given as

$$\sum_{(f,t)\in[k]\times[k]} \omega_{ft} \cdot [M_k(f,t)]_p \;\leq\; \mathfrak{e}_p \qquad \text{for all } p \in E\,. \tag{2.7}$$

The practical driving cycles, for instance the New European Driving Cycle (NEDC) or Real World Driving Cycle (RANDOM), are given by operational profiles which map a time to a gear-velocity combination; cf. Figure 3 and Section 2.3.1. The weights ω_{ft} can be derived from these profiles. The revolution frequency at a certain time can be calculated directly from the current gear/velocity combination.

For the requested engine torque not only the speed but also the acceleration has to be taken into account, along with the car's mass, roll drag and air flow resistance. For a schematic of a discretized driving cycle, cf. Figure 4.

Discrete optimization formulation

In the discrete version of problem (2.4), we approximate the objective function in a similar fashion by replacing again the integral along the curve φ by a weighted sum. To this end, let as above the weight function

$$\Phi\colon [k] \times [k] \to \mathbb{R}_{\geq 0},\; (f,t) \mapsto \Phi_{ft}$$

be the discrete analogue of the curve φ describing how long the curve φ stays in OP_{ft}. Again, Φ may coincide with DC from (2.6) or be chosen to optimize fuel consumption on

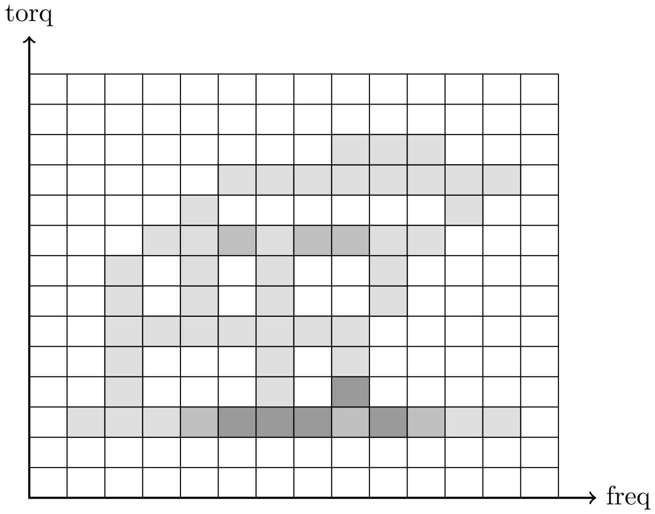


Figure 4: Schematic of a discretized operation field and driving cycle. Resistance times are represented by gray shades. (See also Figures 9 and 10.)

a larger part of the operation field. The optimization objective is now to find a solution map SOL $\in \Omega_k$ by picking for each stack S_{ft} a single representative $d_{ft} = \mathrm{SOL}(f,t)$, so that SOL $\in \Omega_k$ solves

$$\text{minimize}_{M_k \in \Omega_k} \sum_{(f,t)\in[k]\times[k]} \Phi_{ft} \cdot [M_k(f,t)]_{\text{fuel}} \text{ subject to the emission constraint (2.7).} \tag{2.8}$$

Note that $M_k \in \Omega_k$ includes the discrete drivability constraint (2.5). In Section 2.4.5 we will demonstrate how to formulate this problem as an integer linear program (ILP).

2.4 Semi-automatic Calibration

Our method consists of two parts, one semi-automatic, and one automatic. Algorithm 1 is a first rough sketch of the automatic part, which will be detailed in the following. As its input it is given the domain U_{ad}, equipped with a grid, and an evaluation oracle for the function F. It returns a solution map SOL: $[k] \times [k] \to \mathbb{R}^m \times \mathbb{R}^n$, SOL $\in \Omega_k$, of representatives of the k-operation field. In practice, the evaluation oracle for F is given by an engine mounted on a test-bench.

The while loop of Algorithm 1, whose description makes up the bulk of this section, terminates if a preliminary solution map $\widetilde{\mathrm{SOL}}\colon [k] \times [k] \to \mathbb{R}^m \times \mathbb{R}^n$ could be extracted from the measured data. This preliminary solution map $\widetilde{\mathrm{SOL}}$ conforms to the emission and drivability constraints. However, for each stack S_{ft} in $k\,\mathrm{OP}$, it contains either a data point d_{ft}, which will then represent S_{ft}, or a placeholder that adds penalties to the total emission; cf. Section 2.4.5. Note that the constraint $M_k \in \Omega_k$ in (2.8) requires, that there are no empty stacks, but our integer linear program (ILP) formulation of (2.8) in 2.4.5, which is used by Algorithm 1, is extended such that it can handle empty stacks by using a placeholder instead. In the second part, the gaps in $\widetilde{\mathrm{SOL}}$ (marked with placeholders) are closed via interpolation-guided measurements and a complete solution map $\mathrm{SOL} \in \Omega_k$ is returned; cf. Section 2.4.6. The penalty values ensure that replacing a placeholder by a measured value can only improve the solution. As the preliminary solution map $\widetilde{\mathrm{SOL}}$ conforms to the given emission constrains, so must the complete solution map SOL, which is thus a solution to problem (2.8).

In the semi-automatic part of our method, engineering knowledge is used to provide a meaningful base set of measurements for Algorithm 1 and thus guide the calibration process. During this so-called basic calibration, the first two steps in Algorithm 1 are repeated for a preset amount of time with several actuators fixed to values that enforce measurements in regions which are known to be critical to any calibration effort, regardless of the specific engine. In particular, low torque regions, which have a significant influence on the overall pollutant emission, are focused on hereby. As this semi-automatic part of our method consists of a subset of the steps in Algorithm 1, we did not dedicate a separate section to its description. Instead, we provide a protocol of the actuator settings and their respective purposes during basic calibration in Section 2.6.1.

Algorithm 1: Engine calibration procedure

```
Input  : admissible domain $U_{\mathrm{ad}}$, grid $G$, admissible range $Y_{\mathrm{ad}}$, data set $\widehat{F}$,
         precision parameter $k$, evaluation oracle $F$
Output: solution map SOL, updated data set $\widehat{F}$
initialization
while no ILP-solution $\widetilde{\mathrm{SOL}}$ found do
    iteration step: adds to $\widehat{F}$
    data cleaning: reduces $\widehat{F}$ to $\widehat{F}_{\mathrm{red}}$
    grid refinement
    check integer linear program from Section 2.4.5 for feasibility with $\widehat{F}_{\mathrm{red}}$ as
      input and extract solution $\widetilde{\mathrm{SOL}}$ if it exists
end
close the gaps in $\widetilde{\mathrm{SOL}}$
return $(\mathrm{SOL}, \widehat{F})$
```

2.4.1 Initialization

The calibration procedure is initialized with the domain U_{ad} equipped with a grid G, the set of noncritical target values Y_{ad}, a (possibly empty) set of data points $\widehat{F}$ and a precision parameter k. The function F is given implicitly; for a given setting of the actuators, the sensors yield the respective function value by means of a physical measurement. We assume that the data points in $\widehat{F}$ reflect true values of F. The domain

$$U_{\mathrm{ad}} = \prod_{i=1}^{m} I_i \tag{2.9}$$

is a product of intervals $I_1, I_2, \ldots, I_m$, where I_i defines the range of variation of actuator i; cf. Section 2.2.4. It may happen that I_i degenerates to a single point. Then the corresponding actuator is called *static*, otherwise it is called *dynamic*.

There are two cases to distinguish. In the first case, the set $\widehat{F}$ of data points is empty. Then, for each dynamic actuator i, the corresponding interval I_i is subdivided into parts of equal length. Otherwise, if $\widehat{F}$ is not empty, then we assume that each interval I_i is equipped with its own subdivision. In either case, the product form (2.9) induces a grid structure G on the domain. If $\widehat{F}$ is empty then this grid G is uniform. In the later stages of the optimization, however, G will become more and more non-uniform.

Each measurand has an interval J_j of noncritical values. For example, the temperature of the test engine has to stay within certain bounds to prevent it from being damaged. Then

$$Y_{\mathrm{ad}} \;=\; \prod_{j=1}^{n} J_j \,.$$

In practice, Y_{ad} is given in part by the physical test engine (e.g., the aforementioned temperature limits) as well as by external factors such as government regulations (e.g., emission limits). Just like F, which is given by the physical test engine, we will assume the set Y_{ad} to stay fixed throughout the whole calibration process. The precision parameter k is dictated by the engine control unit's engine map format.

During the various steps of the calibration, data points will be added to the set $\widehat{F}$. Thus, it may happen that $\widehat{F}$ becomes prohibitively large to perform subsequent steps of the calibration. How to weed out less relevant measurements is the subject of Section 2.4.3 below.

2.4.2 Iteration Step

The basic iteration step can be subdivided into two phases: The generation of a measurement plan, followed by the actual measurement, which is combined with a refinement of the grid.

Generation of the Measurement Plan

For our given grid G and data set $\widehat{F}$, we construct an abstract graph $\mathrm{G} = \mathrm{G}(G, E)$ as follows. The nodes of G are the grid boxes determined by G. Two m-dimensional grid boxes are joined by an edge if their intersection is a grid box of dimension $m-1$. If there are no measurements yet, i.e., if $\widehat{F}$ is empty, then the grid G is uniform, and the graph G is the dual graph of a cubical cell complex. Due to non-uniform refinement, the structure of G will become more complicated.

The graph G is equipped with nonnegative node and edge weights. For a grid box B, we denote by $\#B$ the number of points $(u^q, y^q) \in \widehat{F}$ such that $u^q \in B$. Then the weight w_{b} of a grid box B is chosen as

$$w_{\mathrm{b}}(B) \;=\; \frac{\mathrm{vol}(B)}{\#B + 1}\,, \tag{2.10}$$

which we call the *reciprocal data density* of B. Adding 1 in the denominator prevents division by zero. Further, we define the weight w_{e} of an edge between two adjacent grid boxes B and B' as the *data density* of $B \cup B'$. That is,

$$w_{\mathrm{e}}(B, B') \;=\; \frac{\#(B \cup B')}{\mathrm{vol}(B) + \mathrm{vol}(B')}\,. \tag{2.11}$$

There is some room for adjusting these weights; the general idea is that the weights should reflect the data densities.

The classical way to measure the engine behavior is to take a measurement only once a steady-state is reached after an adjustment of the actuators. Accordingly, this technique is called steady-state, or stationary measurement. A more recent approach is the quasi-stationary measurement, also called sweep-mapping. Here, the actuators are varied slowly and continuously according to a ramp function in order to save measurement time. The output then follows with a little delay, while measurements are taken at a regular frequency; cf. [Ise14, p. 93]. There are several techniques to bound and even compensate the contouring error of quasi-stationary measurements; cf. [Ise14, p. 94ff] and Remark 2.4.2. Throughout the while loop of Algorithm 1 quasi-stationary measurements will be performed exclusively. For these the following concept is crucial.

Definition 2.4.1. *(Measurement Ramp) For two points u^q, $u^r \in U_{\mathrm{ad}}$, we call the set*

$$\left\{ u^q + i \cdot \frac{u^r - u^q}{\ell - 1} \mid i = 0, 1, \dots, \ell - 1 \right\}$$

the measurement ramp *from u^q to u^r with ℓ measurements.*

Zero-entries of $u^r - u^q$ correspond to actuators which are locally static. Let $\delta := |\widehat{F}|$ and suppose that there is an admissible point $u = u^\delta \in U_{\mathrm{ad}}$ where the last measurement took place. If this does not exist, we choose u uniformly at random in the domain U_{ad}. Let B be the grid box containing u. We may assume that B is unique, since u has been constructed in a randomized fashion. The two steps of the generation of the measurement plan are as follows:

I. *Random grid box:* Pick a grid box B' at random with probability

$$\frac{w_{\mathrm{b}}(B')}{W},$$

where $W = \sum_{B \in G} w_{\mathrm{b}}(B)$ is the sum of the reciprocal data densities of all boxes.

II. *Measurement path:* Determine a shortest path $B = B_0, B_1, \dots, B_s = B'$ in G from B to B' using Dijkstra's algorithm with respect to the weights w_{e} in (2.11); cf. Figure 5 and [Coo98, §2.2]. In each box B_q for $q \in [s]$, pick a point $u^{\delta+q}$ uniformly at random. For $1 \leq j \leq s$, connect the points $u^\delta, u^{\delta+1}, \dots, u^{\delta+s}$ by s measurement ramps with ℓ_j measurements each.

This will result in a total of up to $\sum_{j=1}^{s} \ell_j$ new measurements to be added to the set $\widehat{F}$. In our setting, the actuators are varied at a constant speed on each measurement ramp. This speed is adjusted so that at least one actuator is varied at its maximal variation speed; the values for the maximal actuator variation speeds are listed in Table 2.2. The measurement frequency throughout the whole calibration process is set to one measurement per second. As a consequence, both the length and the orientation of the measurement ramps determine the numbers ℓ_j.

Measurement

The actual measurement is combined with an adaptive refinement of the grid. The goal of the iteration is to fill the solution map in a way that the remaining small holes can be closed by extrapolation of the surrounding data. The next step is then:

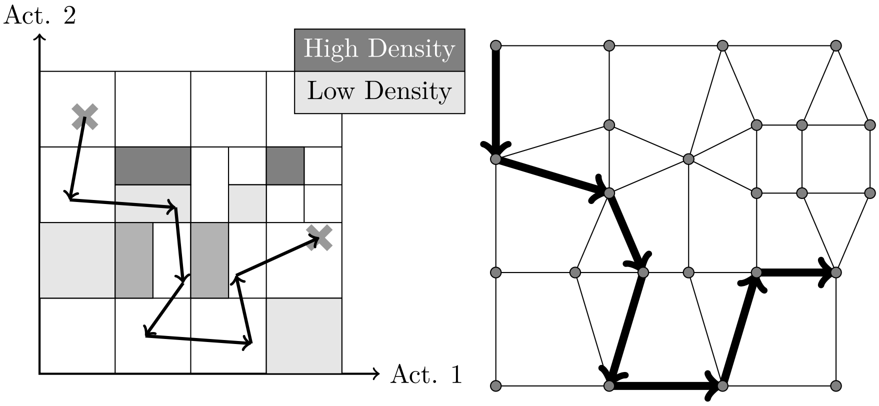


Figure 5: The routing during measurement planning prefers regions of low data density. On the right: Corresponding path in the induced abstract graph.

III. *Measurement:* The actuator settings are varied continuously along the path prescribed by the measurement ramps. This results in a linear ordering of the measurements.

For the recording of the ℓ_j measurements per ramp, several additional aspects have to be taken into account, as explained below. It is important not to store every measurement, since we do not want to store redundant data. Thus, we restrict our attention to *relevant value variations*, i.e., we only store the measurement $F(u^q)$ at a point u^q if the measurement is sufficiently different. Consequently, fewer than $\sum_{j=1}^{s} \ell_j$ data points may be added to the set $\widehat{F}$. Here one can choose among various meaningful distance functions. However, we also want to establish a *minimal measurement frequency.* That is, if we omitted too many subsequent measurements due to the previous rule, then we store the measurement nonetheless.

Moreover, it may happen that a measured value lies outside the admissible range Y_{ad}, i.e., it violates one or more restrictions. In order to *exclude critical values*, the entire measurement is disrupted, and we continue from scratch at (I) with the last valid measurement. Special care is needed for observables with a pronounced latency. The development of these values is extrapolated, and the measurement is rerouted already if the measured values get sufficiently close to the boundary of Y_{ad}.

2.4.3 Data Cleaning

Assume that the measurement phase of the iteration step is complete, i.e., a new set

$$\widehat{F} \;=\; \{(u^1, y^1), (u^2, y^2), \ldots, (u^\delta, y^\delta)\}$$

is available. In Section 2.4.4, we will compute local polynomial fits to the function F using data points in $\widehat{F}$ as interpolation points. At points in which the interpolation is not good enough, the grid is refined further. A fundamental necessity for the good fit of a polynomial approximation is that the interpolation points from which it is generated resemble a random point cloud. This is not the case for $\widehat{F}$, since all points lie on a piecewise linear path given by the sequence of measurement ramps. We thus need to extract a subset $\widehat{F}_{\mathrm{red}}$ from $\widehat{F}$ that is sufficiently generic.

Furthermore, we want to translate the discrete version (2.8) of our optimization problem into an integer linear program which picks the right kind of measurements from which we can then obtain our engine maps; cf. Section 2.4.5 below. However, our measurements need to be preprocessed in order to make such an approach feasible by significantly reducing the amount of data without affecting the accuracy.

As it turns out, we can achieve both goals with a single method, which we call the *adaptive space compressor*. The idea is the following. For the set $\widehat{U} = \{u^1, u^2, \ldots, u^\delta\}$, which is the projection of the set $\widehat{F}$ onto the admissible domain U_{ad}, we define a corresponding *threshold graph*. Its nodes are the points u^q, and there is an edge between u^q and u^r if the Euclidean distance $\|u^q - u^r\|_2$ is below a certain threshold. This threshold depends on the sizes of the two grid boxes which contain u^q and u^r. Then we iteratively remove measurements which have the maximal node degree in the threshold graph until only isolated nodes remain. This *reduced* set is denoted as $\widehat{U}_{\mathrm{red}}$, which yields

$$\widehat{F}_{\mathrm{red}} \;=\; \left\{(u, y) \in \widehat{F} \;\mid\; u \in \widehat{U}_{\mathrm{red}}\right\}.$$

The set $\widehat{F}_{\mathrm{red}}$ has significantly fewer data points compared to $\widehat{F}$. As the threshold for the connection of u^q and u^r by an edge depends on the size of the grid boxes containing them, the distribution of the remaining data points reflects the structure of the grid G. Due to the criteria for grid refinement, which will be introduced in (V) and (VI) below, the structure of the grid G reflects the behavior of F. In particular, this means that the data density is higher in regions where the behavior of F is hard to reconstruct by interpolations. Thus, the adaptive space compressor filters out the relevant information gained by the preceding measurements. Moreover, it breaks up the piecewise linear structure of the data point distribution, leading to local (pseudo-)randomness which is required by the polynomial interpolation.

Note that this reduction does not affect the set $\widehat{F}$, but we rather explicitly keep $\widehat{F}_{\mathrm{red}}$ as a subset. In this way, all our measurements are taken into account for computing the weights according to (2.10) and (2.11) in (I) in the next round of measurements.

Remark 2.4.2. *Since the points u^i are picked in a randomized fashion, the piecewise differentiable map F is differentiable at u^i almost surely. Thus, we may assume that every u^i lies at the center of an open ball on which F is smooth. The grid structure is refined and thus the measurement density increased in areas which are not sufficiently smooth; cf. Section 2.4.4. Consequently, in an individual grid box one can, with high probability, cluster the measured points according to smooth patches of F. As a result, for such clusters the local time constants are identifiable and the techniques for reversing the contouring error by dynamic correction in [Ise14, p. 95f] can be applied successfully, leading to stored measured values that closely approximate the steady state values of the engine.*

2.4.4 Grid Refinement

The purpose of refining the grid G is to accumulate sufficiently many "meaningful" data points in the set $\widehat{F}_{\mathrm{red}}$, such that the k^2 representatives for the solution map can be extracted while observing the emission and drivability constraints. Recall that in our application we have $m \geq 8$ actuators and $n \geq 14$ measurands. This makes computing on a uniform grid infeasible. The price to pay is that it is slightly more involved to determine points u^q in the admissible domain U_{ad} such that the data point (u^q, y^q) adds significant information to our data set $\widehat{F}_{\text{red}}$ and is thus stored. We proceed with the following step:

IV. *Computation of local fit:* We employ a Newton interpolation approach, which requires only a partial recomputation of the interpolation polynomial if some interpolation points are exchanged. Each component function $[F]_i : \mathbb{R}^m \to \mathbb{R}$, where $i \in [n]$, has to be interpolated by a Newton polynomial $L_d[F]_i$, where d is the approximation order. Assuming the $(d+1)$-times continuous differentiability of $[F]_i$ at u, a d-th order polynomial fit requires $N := \binom{m+d}{m}$ interpolation points $\{u^{q_1}, u^{q_2}, \ldots, u^{q_N}\} \subseteq \widehat{U}$. These N points have to satisfy certain spatial relations for the approximation properties of the interpolation polynomials to hold; the interpolation problem is then called *poised*. The formulation of the latter conditions is somewhat technical. We thus omit the details and refer to the literature instead; cf., e.g., [SHI00]. Our data cleaning method in Section 2.4.3 ensures poisedness (with high probability).

    If an interpolation polynomial with N interpolation points which have the required geometrical structure fails to give a d-th order approximation of $[F]_i$ at u, this indicates insufficient smoothness of $[F]_i$ at u, which we will use as a criterion for a local grid refinement. We aim at a second order fit, i.e., $d = 2$, which requires

    $$N \;=\; \binom{m+2}{m} \;=\; \frac{m\,(m+1)}{2}$$

    interpolation points. Our intention is to compare the measured value $y^q \in Y_{\text{ad}}$ at a point $u^q \in U_{\text{ad}}$, where $(u^q, y^q) \in \widehat{F}_{\text{red}}$, with a polynomial interpolation of F in u^q using elements of $\widehat{F}_{\text{red}}$ as interpolation points. The error of such an approximation is minimized if we use the N closest neighbors $\{u^{q_1}, u^{q_2}, \ldots, u^{q_N}\}$ of u^q in U_{ad} as interpolation points, such that

    $$\{(u^{q_1}, y^{q_1}), (u^{q_1}, y^{q_2}), \ldots, (u^{q_N}, y^{q_N})\} \;\subseteq\; \widehat{F}_{\text{red}}\,.$$

Concerning the implementation of the polynomial interpolation: As can be seen in the experimental section below, we will usually have less than $100{,}000$ points in $\widehat{F}_{\text{red}}$. This makes the following brute-force approach feasible. Let $\delta_{\text{red}} := |\widehat{F}_{\text{red}}|$ and assume for simplicity that $q = 1$. Now calculate the squared Euclidean distances of $u^2, u^3, \ldots, u^{\delta_{\text{red}}}$ to u^1. This has a cost of roughly $3 \cdot m \cdot \delta_{\text{red}}$ elementary arithmetic operations. Store these values in the first row of a $(2 \times [\delta_{\text{red}} - 1])$-array and the indices of the corresponding points u^r in the second row ($\approx 2 \cdot \delta_{\text{red}}$ writes). Then determine the N smallest entries of the first row and return the corresponding second row entries.

A simple in-place algorithm to accomplish this is the following: First, determine the N-th smallest element of the first row, which costs one sweep of the array; cf. [CLRS09, p. 183 ff]. Second, sort all columns to the front of the array whose first row entry is smaller than or equal to the N-th smallest first row entry. This costs another sweep of the array. Finally, return the first N second row entries.

Repeating this procedure for all δ_{red} points in $\widehat{F}_{\text{red}}$, we arrive at an approximate cost of $(3 \cdot m + 4) \cdot \delta_{\text{red}}^2 \in \mathcal{O}(m \cdot \delta_{\text{red}}^2)$ elementary operations. In our setting, for up to $100,000$ points and 8 dynamic actuators, this results in a total of about 280 billion elementary operations, which is a fairly insignificant task for modern computers. For a general discussion on the nearest neighbor search in high dimensions we refer to the survey [AI18]. The next step is the following:

V. *Symmetric grid refinement:* To decide whether the grid needs to be refined, we compare the interpolated value, say $\tilde{F}(u^q)$, with $y^q = F(u^q)$. If the deviation exceeds a threshold, then the box B which contains u^q is split into 2^m smaller grid boxes, symmetrically in all coordinate directions.

   The quality of the local fit depends on the differentiability properties of the approximated function. Hence, the symmetric grid refinement increases the measurement density in areas of potential nondifferentiability.

In practice, the static actuators can be ignored for the computation of the local fit and the symmetric grid refinement.

Making the Grid Nonuniform

Recall that ultimately we want to solve the discrete version (2.8) of the constrained optimization problem (2.4), i.e., we want to minimize the fuel consumption subject to the drivability and emission constraints. For the iteration step, the drivability constraints are irrelevant. Therefore we focus on the subset E of measurands corresponding to the emissions. The final output of the calibration will be a solution map $\mathrm{SOL} \in \Omega_k$ represented by k^2 data points $d_{ft} = \mathrm{SOL}(f,t)$, where $f, t \in [k]$, which cover the k-operation field, while satisfying the emission constraints (2.7). In practice, it is a major challenge to find sufficiently many points which satisfy the conditions (2.7) imposed by emission control. This leads us to a second type of grid refinement.

VI. *Asymmetric grid refinement:* Consider the point $u^\delta \in U_{\mathrm{ad}}$ at which the last measurement was performed. Let f and t be indices such that $(u^\delta_{\mathrm{freq}}, y^\delta_{\mathrm{torq}}) \in \mathrm{OP}_{ft}$, where $y^\delta = F(u^\delta)$. If, for any $p \in E$, we have

$$[F(u^\delta)]_p < \min_{(u^q,y^q)\in S_{ft}} ([F(u^q)]_p) ,$$

   i.e., if the emission measurement $[F(u^\delta)]_p$ is lower than any other value on the corresponding stack $S_{ft} \subset \widehat{F}$, a *cross-measurement* is performed. To this end, each actuator is varied individually to determine the direction with the biggest impact on $[F]_p$. Afterwards, the grid box B containing u^δ is split into two congruent (sub-)boxes along the axis corresponding to the actuator whose variation has the biggest impact on $[F]_p$.

Both types of grid refinement do not add data points to $\widehat{F}_{\mathrm{red}}$ directly. However, the grid refinements increase the probability for picking one of the subboxes in (I). This way one can hope to find data points near (u^q, y^q) that add relevant information about F and near (u^δ, y^δ) with better emission values than those currently stored in $\widehat{F}$.

2.4.5 An Integer Linear Program

As advertised in Section 2.3.2, we will now detail how to formulate the discrete optimization problem (2.8) as an integer linear program (ILP) in order derive a preliminary solution map $\widetilde{\mathrm{SOL}}\colon [k] \times [k] \to \mathbb{R}^m \times \mathbb{R}^n$ from the reduced set of measurements $\widehat{F}_{\mathrm{red}}$ that satisfies the objectives listed in Section 2.3.2. This ILP-solution $\widetilde{\mathrm{SOL}}$ is not necessarily a complete solution map SOL, as it may contain some placeholders, which must be replaced in a subsequent interpolation step; cf. Section 2.4.6. For an introduction to the solution of ILPs, see [Coo98], [Sch98], and [Sie01].

For a data point d^q, let S^q be the stack that contains it. As before, let "fuel" be the data point index corresponding to the injected fuel quantity. Then we call the weight

$$\Phi^q := -\frac{\min\bigl\{[d^r]_{\mathrm{fuel}} \;\big|\; d^r \in S^q\bigr\}}{[d^q]_{\mathrm{fuel}}}$$

the *prey value* of d^q. The prey value of d^q is the negative of the quotient of the minimal fuel consumption among all data points in S^q, the stack containing d^q, over the fuel consumption at d^q. As such, the absolute value of any prey value is smaller than or equal to 1. Ideally, a prey value equals -1.

Further, for each index $q \in [\delta_{\mathrm{red}}]$, where $\delta_{\mathrm{red}} := |\widehat{F}_{\mathrm{red}}|$, we introduce a binary decision variable $s^q \in \{0,1\}$, which indicates whether the data point d^q is part of the ILP-solution $\widetilde{\mathrm{SOL}}$. The objective function of our integer linear program can now be written as

$$\text{minimize} \sum_{q=1}^{\delta_{\mathrm{red}}} \Phi^q\, s^q\,, \tag{2.12}$$

while conforming to the constraints (2.13), (2.14), and (2.15), which are described in detail below. It was already noted in Section 2.3.1 that there exists a wide array of meaningful weight functions Φ (and their continuous counter-parts φ). The above choice for Φ, the prey values, ensure that the ILP-solver picks for all operation points the data point with the least possible fuel consumption among all feasible choices.

The stack constraint

The solution $\widetilde{\mathrm{SOL}}$ of our ILP should contain at most one element from each stack S_{ft}. This condition is reflected in the *stack constraint*

$$s_{ft} \;+ \sum_{d^q \in S_{ft}} s^q \;=\; 1 \quad \text{for all } \;(f,t) \in [k] \times [k]\,, \tag{2.13}$$

where s_{ft} is a *stack decision variable*; it is 1 if it is not possible to choose a data point for stack S_{ft} and 0 otherwise. The nonzero s_{ft} are the placeholders we mentioned above. They will be used in (2.15) to assign the penalty values to stacks that contribute no data point to the ILP-solution $\widetilde{\mathrm{SOL}}$.

Formalizing the drivability constraint

To avoid engine damage, the discrete drivability constraint (2.5) ensures, that the variation speeds of all m actuators are constrained individually by nonnegative constants Δ_a

for $a \in [m]$. To write this condition as a constraint in an ILP, let $d^q \in S_{ft}$, and d^r be a data point in either neighboring stack $S_{f\pm1,t}$ or $S_{f,t\pm1}$. Then the *drivability constraints* (2.5) can be expressed as

$$s^q \ + \ s^r \ \leq \ 1 \qquad \text{if} \quad \big|[d^q]_a - [d^r]_a\big| \ \geq \ \Delta_a \quad \text{for any} \ \ a \in [m] \,. \tag{2.14}$$

Note that this is a secant constraint, since the data points compared are contained in neighboring stacks.

Formalizing the emission constraint

The emission condition (2.7) describes the upper bound of several emission test cycles, e.g., maximal `NOx` production. Additionally, the current output of a pollutant $p \in E$ is restricted by the boundaries of the respective interval of noncritical values

$$J_p \ = \ [\underline{e}_p, \overline{e}_p] \,.$$

In the following, the weights ω_{ft} represent the mean resistance time of the rectangles OP_{ft} in the given test cycle. For simplicity of notation, the mean resistance time (cf. Section 2.3.2) on the rectangle corresponding to the stack S^q containing d^q is denoted by ω^q. Then the *emission constraint* can be formulated as follows:

$$\sum_{q=1}^{\delta_{\mathrm{red}}} \omega^q \, [d^q]_p \, s^q \ + \sum_{(f,t)\in[k]\times[k]} \omega_{ft} \, \overline{e}_p \, s_{ft} \ \leq \ \mathfrak{e}_p \qquad \text{for all} \ \ p \in E \,. \tag{2.15}$$

The second summand of the left hand side of the inequality serves as the above-mentioned penalty term which compensates for stacks that contribute no data point to the ILP-solution $\widetilde{\mathrm{SOL}}$.

Remark 2.4.3. *We want to determine the size of our ILP. There are* k^2 *equalities arising from the stack constraints. Further, it has* $m \cdot 2 \cdot k \cdot (k-1)$ *drivability constraints and* $|E|$ *linear emission inequalities. There are* k^2 *stack decision variables and* δ_{red} *decision variables* s^q*, one for each data point in* $\widehat{F}_{red}$*. In the situation of the simulation data presented below, where* $k = 16$*, and* $m = 8$*, this results in* 256 *equalities and* $3840 + 8$ *inequalities which constrain the ILP-solution* $\widetilde{\mathrm{SOL}}$*. Moreover, there are* 256 *stack decision variables. The size of* $\widehat{F}_{red}$ *naturally varies throughout the calibration process, and may range between* 10.000 *and* 100.000*.*

2.4.6 Closing the Gaps in $\widetilde{\mathrm{SOL}}$

The calibration algorithm traces the behavior of the map F, given by the physical test engine, by constructing a sequence of (one-dimensional) measurement ramps through the domain U_{ad} which is high-dimensional, as is the range of F. Naturally, this approach cannot produce a sufficient coverage of U_{ad}. In particular, often several stacks S_{ft} of the k-operation field will contain no data point that contributes to the solution $\widetilde{\mathrm{SOL}}$ of the integer linear program.

For each such non-contributing stack S_{ft} we proceed as follows. First, we construct a local model of F as follows. If S_{ft} lies in the interior of the operation field, we pick

$N = m(m+1)/2$ data points

$$\{(u^{q_1}, y^{q_1}), (u^{q_2}, y^{q_2}), \ldots, (u^{q_N}, y^{q_N})\}$$

from neighboring stacks and compute a Newton type polynomial that interpolates the points $\{u^{q_1}, u^{q_2}, \ldots, u^{q_N}\}$ as done in the local fit step IV. If an empty stack S_{ft} lies on the boundary of the operation field, then the local model is computed by calculating secants of neighboring data points in the nearest stacks and extending them linearly.

Second, we find, e.g., via Newton's method, some point $\tilde{u} \in U_{\mathrm{ad}}$ such that under our local model of F we have $(\tilde{u}_{\mathrm{freq}}, \tilde{y}_{\mathrm{torq}}) \in S_{ft}$. In a ball about $\tilde{u}$ we perform randomized measurements, e.g., using a normal distribution centered at $\tilde{u}$, until we find a point u with $(u_{\mathrm{freq}}, y_{\mathrm{torq}}) \in S_{ft}$ and which satisfies the drivability constraints. Due to the continuity of the manifold map there must exist a neighborhood of such points. The data point $d_{ft} := (u, y)$ is then added to the solution as the representative of the stack S_{ft}. Our way of picking the d_{ft} ensures that the completed solution map conforms to the drivability constraint and we thus have $\mathrm{SOL} \in \Omega_k$.

For pollutant p the penalty value is $\overline{e}_p$, the upper bound of the interval $J_p = [\underline{e}_p, \overline{e}_p]$ of noncritical values. Hence, any recorded value of pollutant p, by which the penalty value is replaced, is smaller than or equal to $\overline{e}_p$. As a consequence, the total emission of a complete solution map SOL cannot exceed that of $\widetilde{\mathrm{SOL}}$ and must thus conform to the emission constraint if the preliminary solution does. Since we also have $\mathrm{SOL} \in \Omega_k$, as noted above, the so-completed solution map does indeed solve problem (2.8).

2.5 AVL Engine Model

For our experiments in Section 2.6, we replace the test-bench with a model of a diesel engine with turbo charger, pilot injection and variable turbine geometry. The latter has been developed in cooperation with AVL GmbH and is based on measurements on a compression ignition/diesel engine. Below we will give a brief overview of the effects of the 8 actuators and 18 measurands that are simulated, thus indicating the scope of the simulation. For a detailed description of the AVL model's derivation, see [Bur15, p. 69ff] and [Ve13].

2.5.1 Actuators of the AVL Engine Model

Revolution frequency of the crankshaft (RF) The crankshaft converts the reciprocating motion of the cylinders into a rotational motion. In modern four-stroke engines every cylinder fires once for every two revolutions of the crankshaft. The revolution frequency is a controlled actuator, as discussed in Section 2.3. It stands out among the other actuators since it provides one coordinate axis of the operation field.

Injected fuel quantity (IF) In contrast to a spark-ignited engine, the injected amount of fuel is the most important actuator. More precisely, the injection process is crucial in the application process of diesel engines. The injection process is given by several pre/pilot-injections, a main injection and post-injections. Typically, the engine torque is mainly determined by the main-injection. This actuator defines the total amount of fuel

Table 2.2: Actuator-intervals during various calibration runs. $\Delta^{\max}$ denotes the maximal variation speed of the actuator in respective units per second. LT/HT = low/high torque, FR = free variation of torque; cf. Sections 2.6.1 and 2.6.2.

Act.	Unit	$\Delta^{\max}$	Basic, LT	Basic, HT	Basic, FR	Full Calibration
`RF`	$\frac{1}{\text{min}}$	10	1000–2600	1000–2600	1000–2600	1000–2600
`IF`	$\frac{\text{mm}^3}{\text{cycle}}$	0.1	6–10	50–60	6–60	6–60
`RP`	hPa	100	295677 / 405677	295677	295677	295677–1126537
`AF`	$\frac{\text{mg}}{\text{stroke}}$	5	300	300	300	275–991
`TG`	int	1	30	30	30	30–85
`MT`	° CA	0.2	0 / 10	0	0	0–10
`PI`	$\frac{\text{mm}^3}{\text{cycle}}$	0	1	1	1	1
`PT`	μs	10	1540	1540	1540	1540–2565

per cycle. In this simple model, the injected fuel volume is divided into one pilot and the main injection.

Pressure in the common rail system (`RP`) In contrast to solenoid-controlled unit injector elements, the pressure is generated by a central fuel and high pressure pump. The fuel injectors are opened and closed by piezo elements.

Air filling (`AF`) Similar to a spark ignited engine, an air valve controls the amount of air which contributes to the combustion process. In this model the amount of air is given directly in mass per piston stroke.

Turbine geometry (`TG`) Modern turbochargers do not have a static turbine geometry. Variable-turbine-geometry turbochargers are able to tune the angle of the turbine blades in order to increase the amount of boost. Alternative setups are given by static turbines with waste gates. Waste gates are applied to reduce the amount of exhaust gas that accelerates the turbine, so the amount of boost can be controlled. In our model the geometry is given as a value between 30 and 85.

Main timing (`MT`) The main timing is comparable to the spark timing of Otto-engines. It defines the start timing of chemical reactions in the crankshaft angle of the main injection. Similar to the Otto engine, the pressure rise is delayed by the ignition delay.

Pilot injection (`PI`) and pilot timing (`PT`) Pilot injection works in tandem with pilot timing to achieve a complete burning of the fuel, which in turn also drastically reduces the emission of $\mathtt{NO_x}$ gases. The pilot injection increases the temperature of the combustion chamber, thus when the main injection occurs the fuel is sent into a chamber which already is at a higher temperature than its autoignition point. This especially facilitates the fuel burning at lower speeds.

2.5.2 Measurands of the AVL Engine Model

Torque The produced torque of the engine. The revolution frequency and the engine torque define the power level of the ICE. It stands out among the other measurands since it provides the second coordinate axis of the operation field.

Fuel mass flow The amount of fuel which is delivered to the cylinder per hour.

Carbon monoxide, hydrocarbons, nitrogen oxides, soot The momentary and integral exhaust emission of pollutants, given in parts per million and emission per hour.

Indicated mean pressure The average pressure over a cycle in the cylinder.

Lambda Lambda displays the chemical partitioning of the fuel. Spark-ignited engines run on lambda around 1. The combustion limits are 0.6 and 1.6. The lambda value for compression ignition engines is much higher, up to 20.

Manifold pressure The manifold pressure measures the absolute pressure in front of the intake channel. The pressure depends on the absolute environmental pressure and the boost level of the turbocharging system.

Boost pressure The boost pressure is created by the turbocharging unit. It depends on the turbine geometry setting and the current combustion behavior.

Maximal cylinder pressure The maximal point of the cylinder pressure sequence. Every engine has a specified maximal cylinder pressure, in order to avoid damaging of the devices. Typically, the maximal pressure is about 160 bar.

Manifold temperature The manifold temperature measures the temperature in the intake channel.

Critical temperature The critical temperature is defined as the temperature of the burn zone when the exhaust valves open. In case of bad timings, the fuel has not been consumed completely, which results in the release of flames to the exhaust manifold, catalysts and turbocharger. It indicates damages to sensitive parts of the engine setup.

Specific fuel consumption The current power level of the engine over the current fuel consumption.

2.6 Optimizing the AVL Engine Model

As explained in Section 2.4, we subdivide the engine optimization into two phases. During the first phase, which we call *basic calibration*, measurements are taken for a prescribed amount of time in regions which are known to be critical to any calibration effort, regardless of the specific engine. The basic calibration is concluded by passing the obtained data to Algorithm 1, which is run without the emission constraint (2.15) in the ILP. If constraint (2.15) is omitted, there always exists a feasible solution $\widetilde{\text{SOL}}$ to the ILP, which may contain several gaps, though, due to the drivability constraint and empty stacks. Hence, the while loop exits immediately and the gaps in $\widetilde{\text{SOL}}$ are closed.

In the second phase, which we call *full calibration*, a solution map conforming to all constraints is obtained by building on and refining the preliminary solution map of the first phase. During full calibration, Algorithm 1 runs automatically and without time limit until it returns a complete solution map SOL.

One could, in theory, omit the first phase and turn the calibration procedure into a fully automatic one by letting all actuators range freely on their whole domain right from the start. However, carefully applying engineering-knowledge in the first phase by bounding some actuators, fixing others, and ignoring the very restrictive emission constraints results in the quick gathering of a meaningful base set of data which then merely has to be refined in the sequel. In our experience, this approach considerably reduces the number of measurements required.

Throughout this section the precision parameter k is set to 16, and this yields 256 representative data points for the solution map SOL. This determines the shapes of Figures 7ff.

Remark 2.6.1. *In the test bench scenario, measurements are taken, though not necessarily stored, in regular time intervals. A realistic frequency is one measurement per second. We take this frequency as the basis of our translation of the number of measurements in the simulation into real-world time. Below, we state the costs of our method in real-world time, as all computation times occurring during the different steps of Algorithm 1 are negligible in comparison to the days, or even weeks, that the physical experiments on a test bench can take.*

2.6.1 Basic Calibration

During basic calibration, only two actuators are dynamic, `IF` and `RF`. The other six actuators are static throughout. The phase is subdivided into three runs of several hours. Here a *run* means the following: The intervals of the dynamic actuator `IF` and the values of the static actuators are reset. Then the algorithm steps in Sections 2.4.2–2.4.4 are repeated for a preset amount of time. The revolution frequency is set to vary over its full range of 1000–2600 revolutions per minute throughout all three runs. For a full account of the applied settings, see Table 2.2.

A good foundation for the optimization process is a well-defined low and high torque boundary region. In a compression ignition engine the generated torque is roughly proportional to the injected fuel. During the first run we measure the low torque region by limiting the `IF`-interval to 6–10 mm^3/cycle, while the static actuators are set to values that support the creation of low torque operation points; cf. Table 2.2. This run takes 6

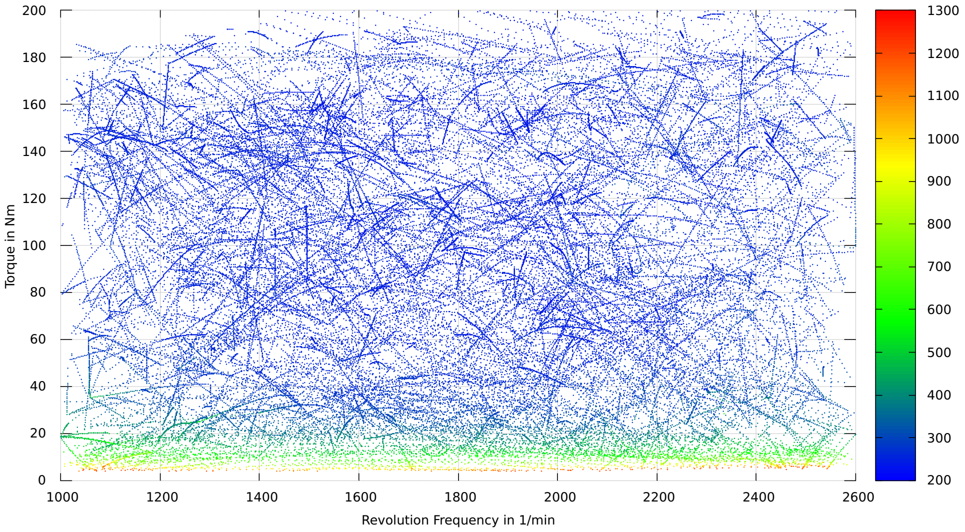


Figure 6: Operation field saturation after basic calibration, before closing of gaps.

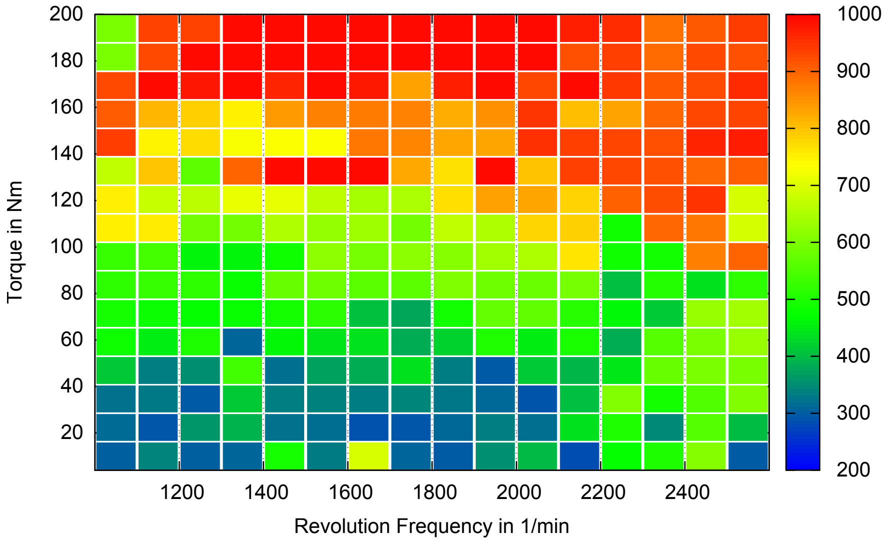
hours. The low torque region is measured for a second time span of 6 hours with slightly modified settings, i.e., `MT` = 10.0 and `RP` = 405677 hPa. We measure the low torque region twice because it represents, in particular, the situation during startup, where low engine temperatures lead to unclean combustion, which causes high `HC` and `CO` emissions; cf. for example [MP15]. Hence, a detailed image of the engine behavior in this area of the operation field is desirable.

Afterwards, the `IF`-interval is reset to 50–60 mm^3/cycle, so that the high torque regions can be measured. The static actuators are set to values that support high torque operation points; cf. Table 2.2. Again, the run takes 6 hours.

Finally, the lower and upper bound of `IF` are removed, in order to get a picture of the remaining region of the solution map. Therefore, the static actuators are set to midrange values; cf. Table 2.2. This region is measured for 6 hours, too. After 24 hours, we get a coarse picture of the engine behavior.

As indicated above, the so-obtained data are used as a base set for a first run of Algorithm 1, albeit with a deactivated emission constraint. Due to the omission of constraint (2.15), the while loop exits immediately as a feasible solution $\widetilde{\text{SOL}}$ to the ILP can always be found in this case. Due to empty stacks and the drivability constraints, which are still active, there may still be gaps in $\widetilde{\text{SOL}}$, though. These are closed subsequently by interpoation-guided measurements; cf. Section 2.4.6. For the comparison of an engine map, i.e., a single component of a solution map, that respects the drivability constraint to one that does not, see Figure 7.

It took an additional 45.8 hours to close all measurement holes. Figure 6 depicts the saturation of the operation field before this completion. The average fuel consumption for 100 kilometers of the so-derived preliminary solution is 4.65 liters in the NEDC. Since



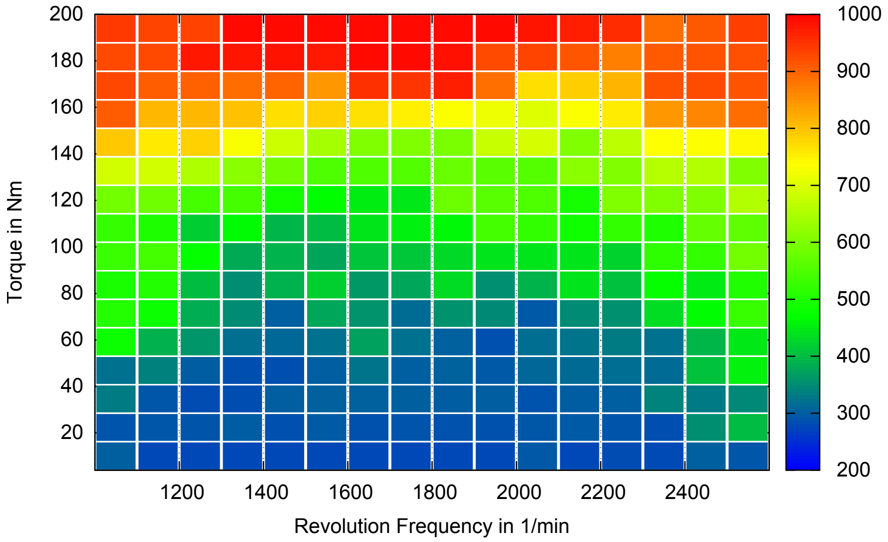


Figure 7: Top: Solution for actuator `AF` without drivability constraint. Bottom: `AF` settings of EURO 5 solution respecting the drivability constraint.

this optimization does not take exhaust emissions into account, the cycle integrals of `CO`, `HC` and $\mathtt{NO_x}$ are rather high: 3.68, 1.72 and 5.26, respectively, per NEDC. The solution map obtained in the first calibration phase serves as the basis of the full calibration.

2.6.2 Full Calibration

During the full calibration phase the only static actuator is pilot injection. All others range over their whole respective domains. With these presets, Algorithm 1 runs automatically and without predefined time limit until it returns a solution map SOL.

The optimization is performed, first, for the NEDC, then for the RANDOM cycle. An NEDC solution map that conforms to the EURO 4 norm is obtained after 68.51 hours, one that conforms to EURO 5 after 88.41 hours. It turns out that the NEDC EURO 5 solution map conforms to all EURO 4 constraints with respect to the RANDOM cycle. That is, it is also a EURO 4 solution map for the latter. A EURO 5 solution map is obtained after 107.55 hours of measurement. For the emission constraints corresponding to the different EURO norms, see Table 2.1.

To lend some context to these numbers, we performed the same calibrations using uniform grids. To derive a solution map of equivalent quality, i.e., a solution map conforming to all constraints given by the different EURO norms, the uniform grid approach consistently required 15 to 20 times as many measurements, whereby we again mean performed measurements, not stored ones. This translates into a real-world measurement time of weeks instead of days for a solution map of comparable quality.

Remark 2.6.2. *Since the AVL model only has a measurand for particle mass, but not for the number of particles, the determination of a EURO 6 solution map lies outside of the model's scope. However, the step from EURO 5 to EURO 6 merely adds an item to the list of emission constraints. This poses no fundamental challenge to our method which is scalable with respect to the number of pollutant limits. Naturally, adding further constraints will increase the measurement time though.*

Description of Figures

Figure 8 displays the lower $\mathtt{NO_x}$ output on the better part of the operation field for the RANDOM calibration in comparison to a calibration for the NEDC.

Figures 9 and 10 illustrate a typical feature of engine calibrations fitted to specific driving cycles. These essentially subdivide the solution map into two parts. One covered by the cycle, where emissions are optimized, and one where they are not. This leads to emission values being 20–100 times higher on the part of the solution map that is not covered by the driving cycle. The NEDC covers only a fraction of the operation field, while the RANDOM cycle covers more than half of it. In Figure 11 one can observe that the solution map for the RANDOM cycle displays 10–15 % higher values for specific fuel consumption on most points of the operation field than the corresponding operation points of a solution for the NEDC. One can, of course, enforce the selection of data points with lower emission values on the part of the operation field not covered by a driving cycle as well, but at the expense of a higher fuel consumption.

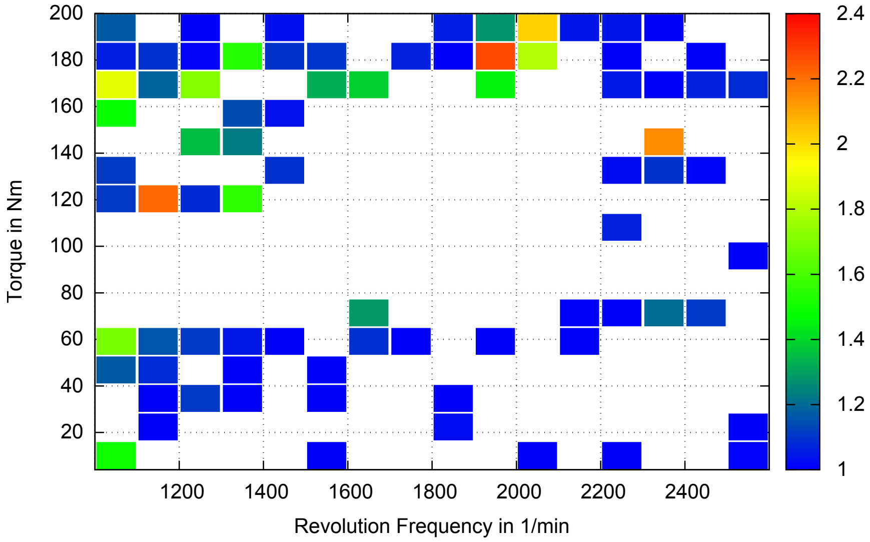


Figure 8: Operation points where calibration for RANDOM cycle yields higher $\mathtt{NO_x}$ emission than in NEDC. Colors represent the ratio of the emission values.

2.7 Conclusion

In this article, we have described a semi-automatic approach to calibrate and optimize internal combustion engines with several actuators and sensors. The side constraints are limits given by safety or technical requirements, bounding the variation speed of actuators and ensuring emission bounds on given driving cycles. Our method automatically performs refinements of measurements, thus focusing the effort of the measurements around regions of strongly nonlinear or even nonsmooth behavior of the engine. That is, it automatically identifies neighborhoods of anomalous engine behavior and maps them in appropriate detail. For the so-obtained data an optimal calibration solution is computed.

The output of the algorithm is a solution map SOL which consists of actual measurements and thus reflects the exact behavior of the engine for the given settings, as opposed to indirectly derived actuator setting obtained via modeling or interpolation. This results in improved values for pollutant emission and fuel consumption near strongly nonlinear or even nonsmooth regions of the admissible domain.

In our experiments, we demonstrated the practicability of the adaptive meshing methodology, showing a significant speed-up in the measurement time (from weeks down to days) in comparison to uniform grids, without a loss of overall quality. Moreover, the resulting solution maps respect the emission constraints of EURO 4 and 5 norms. We would like to stress that in our method it is easy to take into account further emission constraints such as, e.g., the number of emitted particles for the EURO 6 norms.

The next interesting step will be to test the method on an actual combustion engine. It is our expectation that the experimental findings of this work will transfer well to the real-life setting which is the engine test bench.

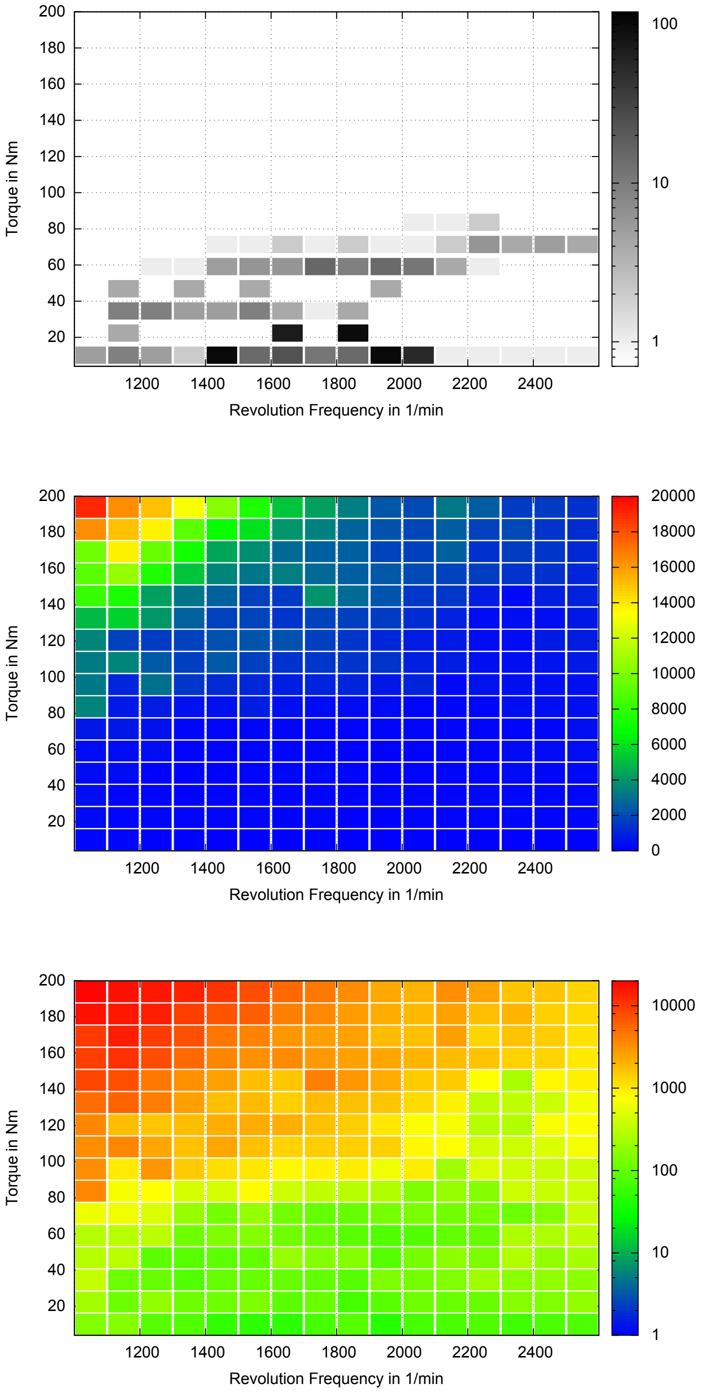
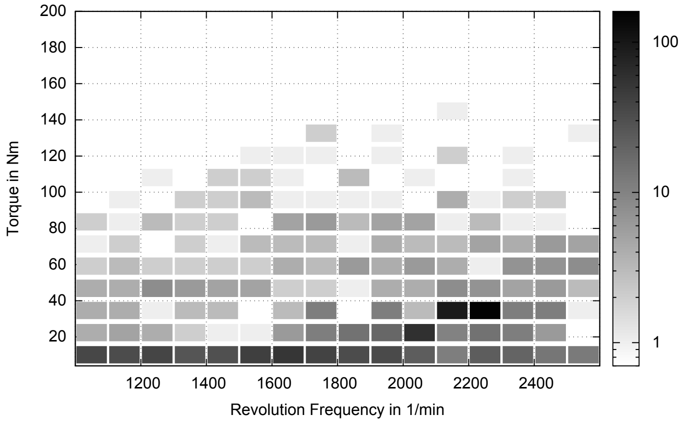
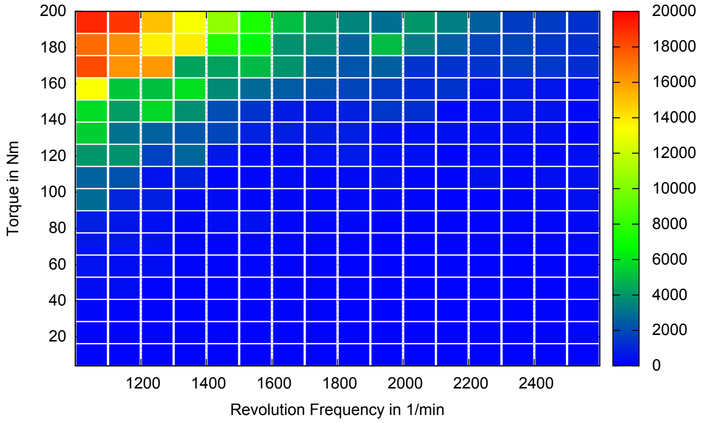


Figure 9: Top: Weighting table of the NEDC. Middle: Resulting $\mathtt{NO_x}$ emissions (in g/h) after calibration for the NEDC. Bottom: $\mathtt{NO_x}$ emissions with logarithmic scale.





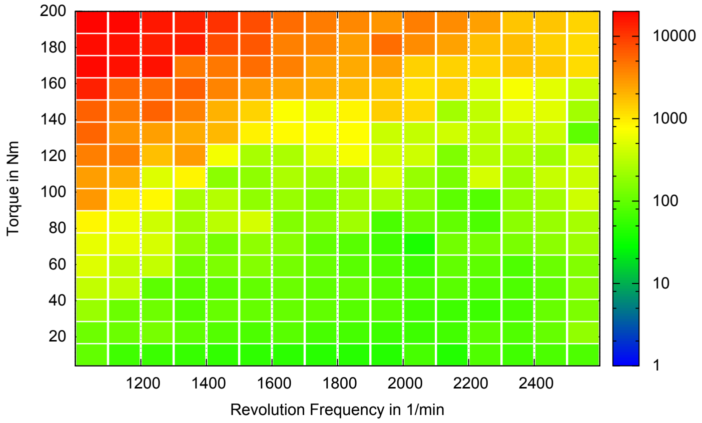


Figure 10: Top: Weighting table of the RANDOM cycle. Middle: Resulting $\mathtt{NO_x}$ emissions (in g/h) after calibration for the RANDOM cycle. Bottom: $\mathtt{NO_x}$ emissions on a logarithmic scale.

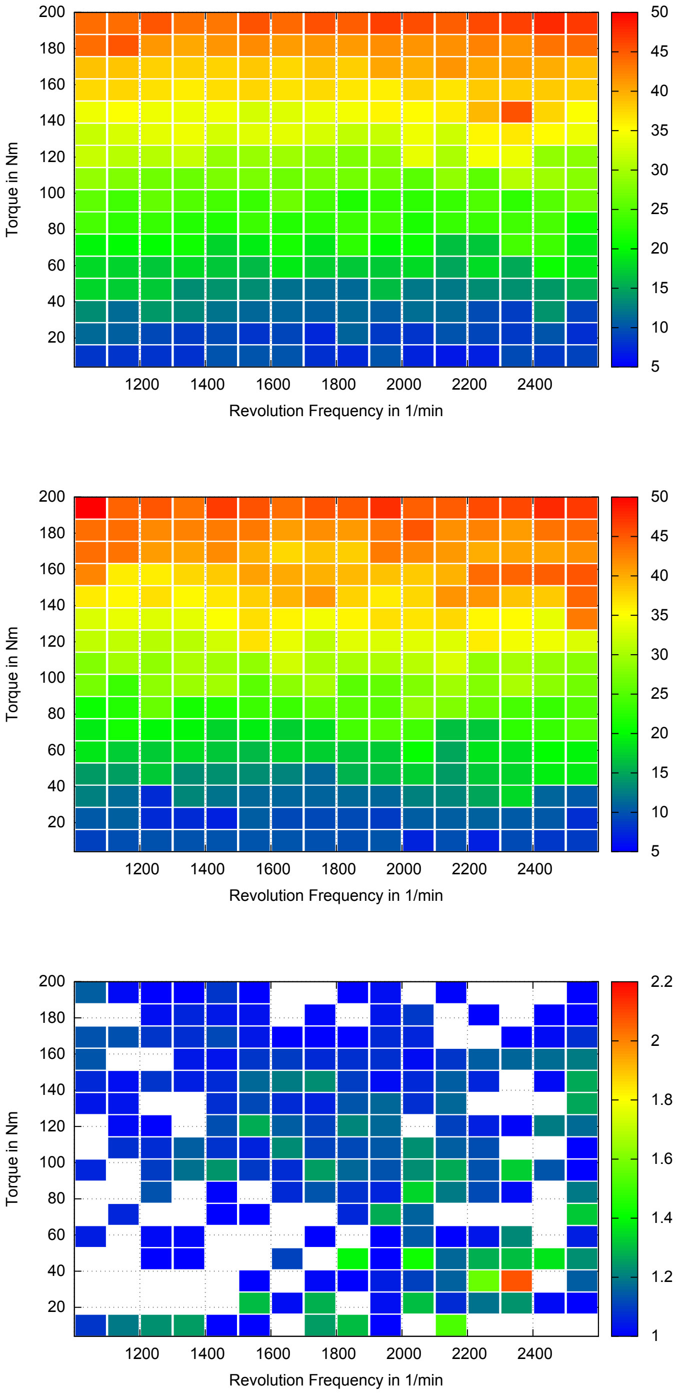


Figure 11: Actuator `IF` after calibration for the NEDC (top) and RANDOM cycle (middle). Bottom: Operation points with higher values in calibration for RANDOM cycle than for NEDC. Colors represent ratio of consumption values.

Chapter 3

Edge-unfolding nested prismatoids

This chapter is taken from the article “Edge-unfolding nested prismatoids” by Manuel Radons. An extended abstract of this work has been published in the proceedings of the Symposium on Computational Geometry: Young Researchers Forum 2022 [Rad22].

3.1 Introduction

The question whether every 3-polytope has a net, that is, whether it is possible to cut it along some spanning tree of its edge graph so that the resulting connected surface may be unfolded flat into the plane without self-overlaps, can be dated back to the “Painter’s Manual” by Albrecht Dürer [Dü25]. It is thus often referred to as Dürer’s Problem.

A polytope that has a net is called edge-unfoldable, or simply unfoldable. It was proved by Ghomi that every polytope is unfoldable after an affine stretching, which implies that every combinatorial type of polytope has an infinite number of unfoldable realizations [Gho14]. O’Rourke established the unfoldability of nearly flat, acutely triangulated convex caps [O’R18, O’R17]. A convex cap is a polytope C which has a designated facet F so that the orthogonal projection of $C \setminus F$ onto F is one-to-one. An acute triangulation is a triangulation so that every interior angle of every triangle is smaller than $\pi/2$. A recent negative result, which Barvinok and Ghomi distilled from a highly original but flawed preprint of Tarasov [BG17, Tar08], concerns the existence of counterexamples to a more general form of Dürer’s problem which considers cuts along so-called pseudo-edges, which are geodesics in the intrinsic metric of a polytope. Another generalized form of Dürer’s problem concerns unfoldability of non-convex polytopes which are combinatorially equivalent to a convex 3-polytope. There are several ununfoldable families of such polytopes known, cf. [Grü02, Tar99, DDE20].

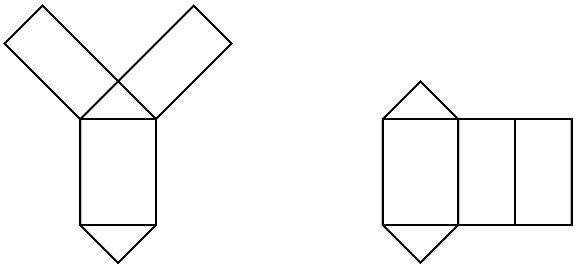


Figure 12: Petal and band unfolding of a prism over a triangle.

3.1.1 Unfolding Prismatoids

A prismatoid P is the convex hull of two convex polygons A and B that lie in parallel planes, say H_A and H_B. We will refer to A and B as the top and base of P, respectively. If all lateral facets of a prismatoid are trapezoids, it is called a *prismoid*. In this case corresponding top and base edges are parallel. The set of lateral facets of a prismatoid is called its *band*. There are two natural ways to unfold primatoids, the *band unfolding*, and the *petal unfolding* [O'R12], cf. Figure 12.

A band unfolding cuts one lateral edge, and unrolls the band into the plane as one connected patch, while A and B are left attached to the latter along one suitable edge each. Every prismoid has a lateral edge e so that the band, if cut along e, can be unfolded without self-intersections [Alo05, ADL$^+$08]. But there exist prismoids so that hat every placement of the prismoid top overlaps with every of its band unfoldings [O'R07].

In a petal unfolding either A or B is a designated facet to which all lateral facets are left attached. Assume that the designated facet is B. Then for each vertex b_i of B exactly one lateral edge adjacent to it is cut. The so-resulting *petals* are unfolded into the plane while A is left attached to this unit along a single suitable edge. O'Rourke proved that every prismoid has a non-overlapping petal unfolding [O'R01]. Smooth prismatoids, which are the convex hull of two smooth convex curves lying in parallel planes, have a petal unfolding as well [BCO04]. Further, several subclasses of prismatoids are known to have a petal unfolding. A nonobtuse triangle is a triangle so that all its interior angles are smaller than or equal to $\pi/2$. A prismatoid has a petal unfolding if all its facets, except possibly its base B, are nonobtuse triangles [O'R12], or if the base is a rectangle and all other facets are acute triangles, or if the top and base are sufficiently far from each other [BDM21].

3.1.2 Main result

A prismatoid is *nested* if the orthogonal projection of A onto H_B is properly contained in B, or vice versa. We prove the following result.

Theorem 3.1.1. *Let P be a nested prismatoid. Then P is edge-unfoldable.*

To this end we apply a combination of the petal and the band unfolding strategies to nested prismatoids. More precisely, we cut the band into two pieces. Crucial in the selection of the band-patches which are left intact is the notion of radially monotone polygonal paths, which was introduced and exploited to great effect in [O'R18].

3.1.3 Content and Structure

In Section 3.2 we will introduce the necessary concepts for our investigation and a few preliminary results. Sections 3.3 and 3.4 contain the proof of our main result. In Section 3.3 we devise a strategy to cut a nested prismatoid into four pieces. In Section 3.4 we establish that these pieces can be glued together into an unfoldable polyhedral surface.

3.2 Preliminaries

We will follow notation and naming conventions established by Ghomi in [Gho14] and O'Rourke in [O'R18, ADL$^+$08] whenever possible. Throughout this work "polytope" means the boundary of a convex 3-polytope which lives in $\mathbb{R}^3$. Consequently, a prismatoid P is the boundary of the convex hull of two polygons $A \subset H_A$ and $B \subset H_B$, where H_A and H_B are parallel affine planes. We will assume without loss of generality that the projection of A to H_B is properly contained in B. We call A the top of P and B its base. Further, we will assume that H_B is the xy-plane embedded in $\mathbb{R}^3$ and H_A, which is a parallel to H_B, has a positive height. Vertices of A and B are denoted a_i and b_j, respectively.

Throughout, polygons are convex unless explicitly stated otherwise. Vertices of an n-gon are enumerated counterclockwise from 1 to n with respect to a viewpoint above the polygon. Let $D \coloneqq \operatorname{conv}(d_1, \ldots, d_n)$, where conv denotes the convex hull of the vertices $d_1, \ldots, d_n$.

A subpath $[d_i, d_{i+1}, \ldots, d_j]$ of the boundary of D is denoted (d_i, d_j). We define the *curvature* at a vertex d_k as the angle spanned by the outward normals of D at d_k. The *total curvature* of a subpath (d_i, d_j) is the sum of the curvatures at its interior vertices. In some sources these quantities are also called the *turn angle* and the *turn*; see, for example, [Alo05]. For brevity we will refer to the total curvature of a path simply as its curvature, if the meaning is clear from the context. If a band piece is bounded by the top-subpath $[a_k, \ldots, a_\ell]$ and the base-subpath $[b_K, \ldots, b_L]$, then we call (a_k, a_ℓ) its top boundary and the curvature of (a_k, a_ℓ) its top curvature. Base boundary and curvature are defined analogously. We use the same letters in upper and lower case to underline the facts that the end vertices of top and base boundary a) correspond to each other, while b) they usually do not have the same index.

We define the *flat prismatoid* P^0 corresponding to P as follows: The lower facet of P^0 coincides with the lower facet B of P. The upper facets of P^0 are obtained as the cells of a subdivision of B which is induced by the orthogonal projection of $P \setminus B$ onto B.

3.2.1 Projections and unfoldings

Let P be a nested prismatoid with top A and base B. Then we denote the orthogonal projection of any subset $C \subset P$ onto H_B by $\tilde{C}$.

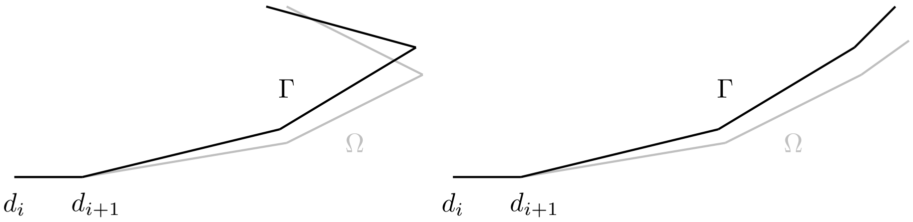


Figure 13: Stretching of paths with curvature $> \frac{\pi}{2}$ (left), resp. $\leq \frac{\pi}{2}$ (right).

It is well-known that any spanning tree of the edge graph of a 3-polytope P induces an unfolding of P into the plane, cf. [DO07, Lem. 2.2.2, p.311]. This unfolding is an isometric immersion which is unique up to rigid motions. Any subset C of a 3-polytope P which has been cut along a spanning tree of its edge graph can be isometrically immersed, i.e., unfolded into a plane as well. We denote the unfolded image of C by $\bar{C}$ and assume the plane to be H_B, the xy-plane embedded in $\mathbb{R}^3$. In particular, $\bar{P}$ denotes the unfolded image of P. In some sources $\bar{P}$ is subscripted with the designation of the spanning tree along which P has been cut. But we will construct only one such tree, hence omitting the index introduces no ambiguities.

We remark that $\bar{P}$ does not self-intersect if and only if its boundary does not self-intersect. The discussion below will establish exactly such a lack of boundary self-intersections.

3.2.2 Radial monotonicity

A polygonal path $\Gamma \subset \mathbb{R}^2$ is called *radially monotone* if traversing it from an endpoint the euclidean distance to that endpoint monotonically increases. In other words, for points $p_i, p_j, p_k \in \Gamma$ the euclidean distance of p_i and p_j is greater than that of p_i and p_k if and only if their intrinsic distances in Γ have the same relation; cf. [O'R18]. Let $\Gamma := (d_i, d_j)$ be a subpath of a convex n-gon $D := \operatorname{conv}(d_1, \ldots, d_n)$ in some plane H, and Ω another polygonal path in H that is obtained from Γ by keeping its first edge, (d_i, d_{i+1}), fixed and decreasing its curvature at all interior vertices, but not decreasing it to 0 or less at any vertex. We then say that Ω is obtained by stretching Γ. In [O'R18] we find the following crucial observation.

Observation 3.2.1. *Let $\Gamma := (d_i, d_j)$ be a subpath of a convex n-gon D in some plane H, and let Ω be obtained by stretching Γ. If (d_{i+1}, d_j) is radially monotone, then Γ intersects Ω only in its first edge (d_i, d_{i+1}), cf. Figure 13.*

Note that for a subpath of the boundary of a polygon a sufficient condition for radial monotonicity is that its curvature does not exceed $\pi/2$.

3.2.3 Flattening of the band

The observation below was stated for the special case of nested prismoids in both [O'R07] and [ADL$^+$08]. We will generalize it to nested prismatoids.

Observation 3.2.2. *Let M be a connected piece of the band of some nested prismoid P and let v be an interior vertex of either its top or its base boundary. Then the curvature at $\bar{v}$, the image of v under the unfolding of M into the plane, is smaller than the curvature at v, but larger than 0.*

First, recall some definitions from the literature. The total angle of a vertex is the sum of the incident face angles. For any convex polyhedron, the total angle is $\leq 2\pi$ with equality if and only if the incident faces lie in a plane (which does not occur in our situation). Let M be a connected piece of the band of some nested prismatoid P which is bounded by the polygonal path $[a_k, \dots, a_\ell, b_L, \dots, b_K, a_k]$. For $i \in \{k+1, \dots, \ell-1\}$, let $\alpha_A(a_i)$ be the incident top angle at a_i, and $\alpha_M(a_i)$ the sum of the incident band angles. Similarly, for $j \in \{K+1, \dots, L-1\}$, let $\beta_B(b_j)$ be the incident base angle at b_j, and $\beta_M(b_j)$ the sum of the incident band angles. Further, define $\alpha_{\bar{M}}(a_i)$ and $\beta_{\bar{M}}(b_j)$ for the unfolded band piece $\bar{M}$ analogously to $\alpha_M(a_i)$ and $\beta_M(b_j)$.

Now let $\bar{M}$ be an unfolding of M into the plane. Due to the assumption that P is nested, we have

$$\beta_B(b_j) \;<\; \beta_M(b_j) \;=\; \beta_{\bar{M}}(b_j) \;<\; \pi\,.$$

Since the total angle at a_i is smaller than 2π, we get

$$\alpha_A(a_i) \;<\; 2\pi - \alpha_{\bar{M}}(a_i) \;=\; 2\pi - \alpha_M(a_i) \;<\; \pi \;< \alpha_M(a_i)\,.$$

That is, the unfolding of (b_K, b_L), the base boundary of M, is a stretching of its orthogonal projection to the plane. Likewise, (a_k, a_ℓ) is stretched by its unfolding. But this is what we needed to show.

3.3 Cutting strategy and placing the top

We now collect some observations about band unfoldings and then derive our cutting strategy from these insights.

3.3.1 Observations about band unfoldings

Let M be a connected piece of the band of a nested prismatoid P, bounded at the top by a path $\Gamma \coloneqq [a_1, \dots, a_k]$ and at the base by a path $\Omega \coloneqq [b_1, \dots, b_K]$. Assume that the pairs of edges (a_1, a_2), (b_1, b_2), and (a_{k-1}, a_k), (b_{K-1}, b_K) are each contained in a lateral trapezoid and are thus parallel. Then by elementary geometry Γ and Ω both have the same curvature. Assume that this curvature is $\leq \pi$. Then, due to Observation 3.2.2, the curvature of $\bar{\Gamma}$ and $\bar{\Omega}$ is smaller than π, while the curvature at every interior vertex of $\bar{\Gamma}$ and $\bar{\Omega}$ larger than 0. Together with the fact that their end edges must be parallel, this implies that $\bar{M}$ does not self-intersect and any line through an edge of $\bar{\Omega}$ induces a closed half plane that contains $\bar{M}$.

Moreover, since the curvature of $\bar{\Gamma}$ is smaller than π, by elementary arithmetic there must exist an edge $(\bar{a}_i, \bar{a}_{i+1})$ so that if its relative interior is removed from $\bar{\Gamma}$, each of the two remaining subpaths of $\bar{\Gamma}$ either consists of a single vertex or has a curvature $\leq \pi/2$ and is thus radially monotone. Hence, Observations 3.2.1 and 3.2.2 imply that if we attach A (thereafter $\bar{A}$) to $\bar{M}$ along $(\bar{a}_i, \bar{a}_{i+1})$, it does not intersect $\bar{M}$ anywhere

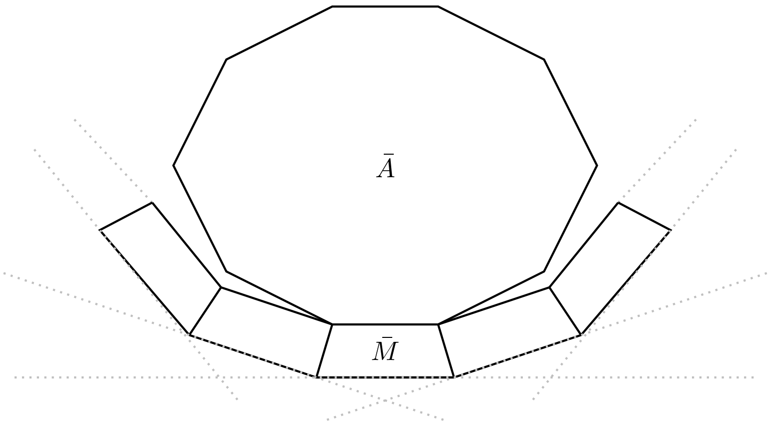


Figure 14: Containing $\bar{M}$ and $\bar{A}$

else. Moreover, any line through an edge of $\bar{\Gamma}$ induces a closed half plane that contains $\bar{A}$. In particular, $\bar{A}$ cannot properly intersect the affine rays emanating from $\bar{a}_2$ and $\bar{a}_{k-1}$ through the edges $(\bar{a}_1, \bar{a}_2)$ and $(\bar{a}_{k-1}, \bar{a}_k)$, respectively. Composing the latter rays with the subpath $[\bar{a}_2, \ldots, \bar{a}_{k-1}]$ yields an unbounded curve, say $\bar{\Gamma}'$, that intersects $\bar{A}$ only in the edge $(\bar{a}_i, \bar{a}_{i+1})$. Since $\bar{\Gamma}$ and $\bar{\Omega}$ are parallel in their ends, every line through an edge of $\bar{\Omega}$ – including the lines through its first edge $(\bar{b}_1, \bar{b}_2)$ and its last edge $(\bar{b}_{K-1}, \bar{b}_K)$ – induces a closed half plane whose interior contains $\bar{\Gamma}'$ and thus also $\bar{A}$, cf. Figure 14. We summarize the relevant aspects of these findings.

Observation 3.3.1. *In the above constellation, $\bar{M}$ does not self-intersect. Moreover, there exists an edge e of $\bar{\Gamma}$ so that if we attach A (thereafter $\bar{A}$) to $\bar{M}$ along e, the resulting flat polyhedral surface, say $\bar{N}$, does not self-intersect and every line through an edge of $\bar{\Omega}$ induces a closed half plane that contains $\bar{N}$.*

3.3.2 Cutting and placing the top

We will now devise a cutting-scheme that recreates the above ideal constellation sufficiently well to harness all its advantages. Let P be a nested prismatoid with top A, base B, and corresponding flat prismatoid P^0. We assume that A is an m-gon with boundary $[a_1, \ldots, a_m]$ and B an n-gon with boundary $[b_1, \ldots, b_n]$.

Now pick any vertex of B. By rotating the indices, if necessary, we can assume that we picked b_1. Let L_1 be the line through the edge (b_n, b_1). We say a line L supports a polygon C in a plane H if it lies in H, has a nonempty intersection with C and C is contained in one of the two closed half planes induced by L. Let L_2 and L_3 be the unique disjoint supporting lines of $\tilde{A}$ which are parallel to L_1, and let L_2 be the one closer to L_1. Further, let L_4 be the unique supporting line of B which is parallel to L_1 and disjoint from it, cf. Figure 15 (left).

We denote by K the smallest index so that b_K is contained in L_4. Further, enumerate the indices of $\tilde{A}$ so that $\tilde{a}_1$ is contained in L_2, but $\tilde{a}_2$ is not and denote by k the smallest index so that $\tilde{a}_k$ is contained in L_3.

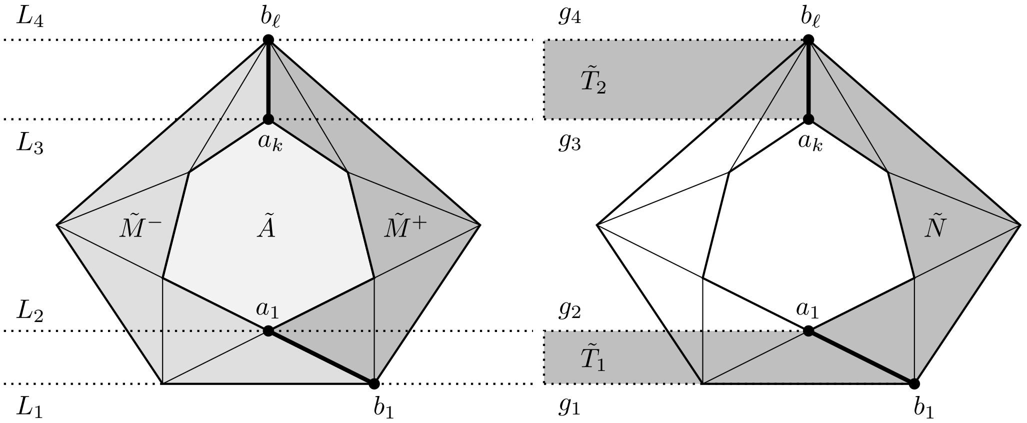


Figure 15: Determining the lateral cut edges; the cut edges are printed fat (left) and gluing trapezoids to the band (right)

Observation 3.3.2. *The vertices a_1 and b_1, as well as a_k and b_K are adjacent. Moreover, the edge (a_1, b_1) lies in a lateral facet which contains the edge (b_n, b_1).*

How do we see this? Pick an arbitrary vertex of the base, say b_i, and let H_- and H_+ be two hyperplanes which contain the lateral facets with base edges (b_{i-1}, b_i) and (b_i, b_{i+1}), respectively. Then the intersections of H_- and H_+ with H_A project orthogonally into lines L_- and L_+ in H_B which are parallel to (b_{i-1}, b_i) and (b_i, b_{i+1}), respectively. Relabel A so that $\tilde{a}_1$ is contained in L_-, but $\tilde{a}_2$ is not, and let k be the smallest index so that $\tilde{a}_k$ is contained in L_+. Then by construction every lateral edge incident to b_i must contain a vertex of the path $[a_1, \ldots, a_k]$, and the first an the last vertices of this path are contained in the lateral facets with base edges (b_{i-1}, b_i) and (b_i, b_{i+1}), respectively.

Now cut the lateral edges (a_1, b_1) and (a_k, b_K), as well as all top and base edges. This dissects P into four pieces, the top A, the base B and two band pieces. We denote the band piece in anticlockwise direction from (a_1, b_1) by M^+ and the one in clockwise direction by M^-.

Next, we recreate our ideal constellation outlined above by embedding M^+ in a strictly larger polyhedral surface that satisfies all conditions which lead to Observation 3.3.1. To this end, let g_1, $\tilde{g}_2$, $\tilde{g}_3$, and g_4 be four parallel line segments of nonzero but otherwise arbitrary length with the following properties.

- They lie in L_1, L_2, L_3, and L_4, respectively.
- They originate in b_1, $\tilde{a}_1$, $\tilde{a}_k$, and b_K, respectively.
- They all extend into the same direction, which is the one where neither $\tilde{g}_2$, nor $\tilde{g}_3$ intersect the interior of $\tilde{M}^+$.

Let g_2 and g_3 be the orthogonal projections of $\tilde{g}_2$ and $\tilde{g}_3$ to H_A and glue the two trapezoids, say T_1 and T_2, which arise as the convex hulls of g_1 and g_2, resp., g_3 and g_4, to M^+ along their common edges (a_1, b_1) and (a_k, b_K), respectively, cf. Figure 15 (right). The so-created polyhedral surface has a top and base curvature of exactly π and both its end facets are trapezoids. Hence, Observation 3.3.1 applies to it – and thus to its subset M^+. Applying the analogous construction to M^-, we get:

Observation 3.3.3. *$\bar{M}^+$ does not self-intersect and there exists an edge e of its top boundary $\bar{\Gamma}$ so that if we attach A (thereafter $\bar{A}$) to $\bar{M}^+$ along e, the resulting flat polyhedral surface does not self-intersect and every line through an edge of the base boundary of $\bar{M}^+$ induces a closed half plane that contains it. Moreover, analogous statements hold for $\bar{M}^-$.*

3.4 Gluing the band pieces to the base

The cutting strategy presented above dissects a nested prismatoid into four pieces, A, B, M^+, and M^-. Moving on, we attach the top A to M^+ along the edge e from Observation 3.3.3. We will denote the polyhedral surface consisting of A glued to M^+ along e by N. There are technical reasons we will explain below to choose M^+ and not M^- for the attachment of the top.

We will now single out two edges along which the three pieces B, M^- and N can be glued together into a polyhedral surface whise unfolding does not self-intersect. We will denote by e_- the edge that connects M^- to B and by e_+ the edge that connects B to M^+. To this end, we distinguish two cases. In the first case there exists a vertex of the base with a curvature $\geq \frac{\pi}{2}$. In the second case no such vertex exists. We will treat the first case in detail and then show how to reduce the second case to the first.

In the first case, label B so that the vertex with curvature $\geq \frac{\pi}{2}$ is b_1 and apply our cutting strategy. Then set $e_- := (b_n, b_1)$ and $e_+ := (b_1, b_2)$. By Observation 3.3.3 we can establish unfoldability in the first case by proving that $\bar{M}^-$ intersects the line through e_+ nowhere except in b_1. Let g be the outward normal ray of e_- at b_1. Since the curvature at b_1 is $\geq \pi/2$, we are done if we can prove that $\bar{M}^-$ intersects g in b_1 and nowhere else. For this it suffices to prove that the base boundary of $\bar{M}^-$ intersects g only in b_1 and its top boundary does not intersect g at all.

By Observation 3.3.2 our cutting strategy ensures the following: If there is more than one lateral edge incident to b_1, then (a_1, b_1) is the first one of them counted in anticlockwise direction and there must exist a lateral facet that contains the edges $e_- = (b_n, b_1)$ *and* (a_1, b_1). (This is not the case for M^+, hence our choice above.) This facet is either a trapezoid or a triangle. If it is not a trapezoid, embed the facet in a trapezoid by taking its convex hull with a point that lies on L_2 and projects into the interior of B, but not into $\tilde{M}^-$. By construction this does not increase the top curvature of $\tilde{M}^-$ beyond π. We will thus assume without loss of generality that the lateral facet containing e_- is a trapezoid.

Since P is properly nested, i.e., $\tilde{A}$ is contained in the interior of B, and due to the curvature at b_1, the reflection of $\tilde{M}^-$ at e_- can intersect g in at most two points, the endpoints of its base boundary, i.e., in b_1 and the reflection of $\tilde{b}_\ell$ at e_-. Let p be a point in either the top or the base boundary of M^-, and p' the reflection of $\tilde{p}$ at e_-. We will show that the distance of $\bar{p}$ to g is larger than or equal to the distance of p' to g, with equality if and only if p is contained in $e_- = (b_n, b_1)$ or (a_m, a_1), that is, if it lies in the top or base edge of the lateral trapezoid that contains e_-.

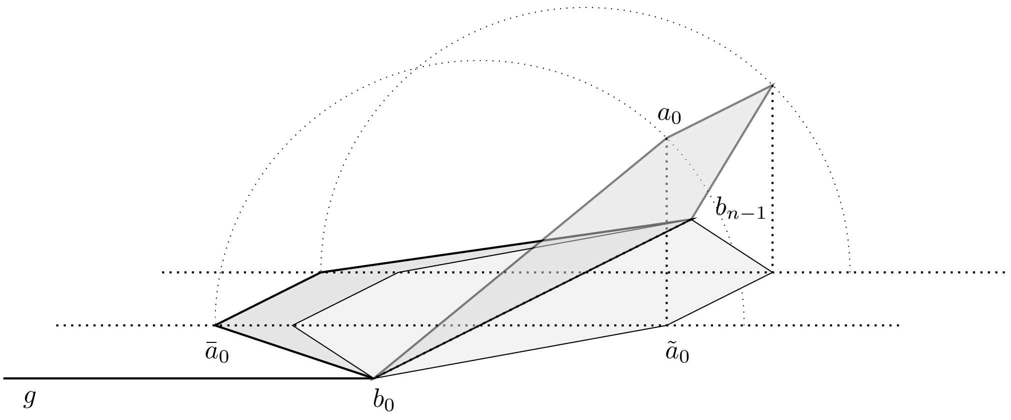


Figure 16: Projection, reflection of projection, and unfolding of lateral trapezoid incident to (b_{n-1}, b_0).

If p lies in e_-, this is clear, because $e_- = \tilde{e}_- = \bar{e}_-$. By elementary geometry, the edges $(\tilde{a}_n, \tilde{a}_1)$, its reflection at e_-, and $(\bar{a}_n, \bar{a}_1)$ are parallel. Moreover, $\tilde{a}_1$, its reflection at e_-, and $\bar{a}_1$ lie on a line which is perpendicular to e_-, cf. Figure 16. Hence, $(\tilde{a}_n, \tilde{a}_1)$, its reflection at e_-, and $(\bar{a}_n, \bar{a}_1)$ can be translated into each other by translating them parallelly to g. This proves the claim for $p \in (a_n, a_1)$.

Now denote by Ω the base boundary of M^-. We can transform the reflection of $\tilde{\Omega}$ at e_-, which we denote by $\tilde{\Omega}'$, into $\bar{\Omega}$ as follows. First, decrease the curvature at $\tilde{b}'_{K+1}$ so that it matches the curvature at $\bar{b}_{K+1}$ in $\bar{\Omega}$. This rotates the edge $(\tilde{b}'_K, \tilde{b}'_{K+1})$ about $\bar{b}_{K+1}$ in clockwise direction (seen from a vantage point above H_B). Since the curvature of $\tilde{\Omega}'$ is $\leq \pi$ and e_- is perpendicular to g, this rotation must increase the distance to g for all points in $(\tilde{b}'_K, \tilde{b}'_{K+1})$, except $\tilde{b}'_{K+1}$. Successively applying this procedure to $\tilde{b}'_{K+2}, \dots, \tilde{b}'_n$ yields the claim. A similar argument can be made for the image of the top boundary with one extra step: First perform the rotations, as above. This yields a curve congruent to $\bar{\Gamma}$. Then shift this curve parallelly to g into $\bar{\Gamma}$, which does not change the distance to g for any point in the translated curve. This completes our proof for the first case that there exists a base vertex with curvature $\geq \pi/2$.

Now assume that B has no vertex with a curvature $\geq \pi/2$. Then by elementary arithmetic there must exist an index i so that the path (b_1, b_i) has a curvature in $[\frac{\pi}{2}, \pi)$. We set $e_+ := (b_{i-1}, b_i)$. Further, set $e_- := (b_n, b_1)$, as in the first case. Let L_- be the line through e_-, L_+ the line through e_+, and p their intersection. Then the curvature of the polygonal path $[b_1, p, b_{i-1}]$ at p is $\geq \pi/2$, and we can prove the claim that $\bar{M}^-$ does not intersect the outward normal ray g of (b_1, p) emanating from p in full analogy to the above claim that $\bar{M}^-$ intersects g only in b_1, where b_1 is a base vertex with curvature $\geq \pi/2$. This completes our proof for the second case that there exists no base vertex with curvature $\geq \pi/2$, and thus completes the proof of Theorem 3.1.1.

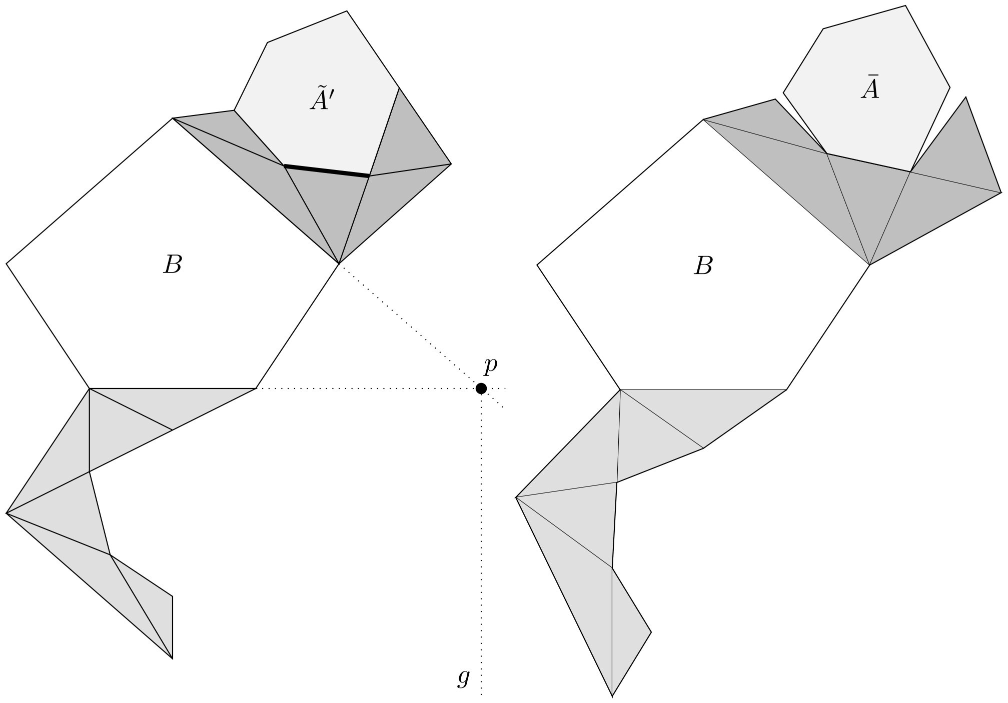


Figure 17: Left: Unfolding of P^0. $\tilde{A}'$, the reflection of $\tilde{A}$, is attached at the fat edge of the reflection of $\tilde{M}^+$. Right: Unfolding of P

Bibliography

[ADL$^+$08] G. Aloupis, E. D. Demaine, S. Langerman, P. Morin, J. O'Rourke, I. Streinu, and G. Toussaint. Edge-unfolding nested polyhedral bands. *Computational Geometry*, 39(1):30–42, 2008. Special Issue on the Canadian Conference on Computational Geometry.

[AI18] A. Andoni and P. Indyk. Nearest neighbors in high-dimensional spaces. In C. D. Tóth, J. E. Goodmann, and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, chapter 43. CRC Press, 2018. 3rd edition.

[AL17] D. Amelunxen and M. Lotz. Average-case complexity without the black swans. *Journal of Complexity*, 41:82–101, 2017.

[Ale50] A. D. Alexandrov. *Convex Polyhedra.* Springer, 2005 (1950).

[Alo05] G. Aloupis. *Reconfigurations of Polygonal Structures.* PhD thesis, McGill University, Montreal, Que., Canada, Canada, 2005. AAINR12794.

[AO92] B. Aronov and J. O'Rourke. Nonoverlap of the star unfolding. *Discrete & Computational Geometry*, pages 219–250, 1992.

[BC08] L. Brugnano and V. Casulli. Iterative solution of piecewise linear systems. *SIAM Journal on Scientific Computing*, 30(1):463–472, 2008.

[BC13] P. Bürgisser and F. Cucker. *Condition – The Geometry of Numerical Algorithms.* Springer, 2013.

[BCO04] N. Benbernou, P. Cahn, and J. O'Rourke. Unfolding Smooth Primsatoids. *CoRR*, cs.CG/0407063, 2004.

[BDM21] V. Bian, E. D. Demaine, and R. Madhukara. Edge-Unfolding Prismatoids: Tall or Rectangular Base. In *CCCG 2021, Halifax, Nova Scotia, Canada, August 10–12*, 2021.

[BG17] N. Barvinok and M. Ghomi. Pseudo-Edge Unfoldings of Convex Polyhedra. *Discrete & Computational Geometry*, pages 1–19, 2017.

[BJP$^+$20] T. Burggraf, M. Joswig, M. Pfetsch, M. Radons, and S. Ulbrich. Semi-automatically optimized calibration of internal combustion engines. *Optim Eng*, 1:73–106, 2020.

[Bur15] T. Burggraf. *Development of an automatic, multidimensional, multicriterial optimization algorithm for the calibration of internal combustion engines.* PhD thesis, TU Darmstadt, 2015.

[CD68] R. W. Cottle and G. B. Dantzig. Complementary pivot theory of mathematical programming. *Linear Algebra and its Applications*, 1:103–125, 1968.

[Chu89] S. J. Chung. NP-Completeness of the linear complementarity problem. *J Optim Theory Appl*, 60:393–399, 1989.

[CLO07] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms.* Undergraduate Texts in Mathematics. Springer, New York, third edition, 2007.

[CLRS09] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Third Edition.* The MIT Press, 3rd edition, 2009.

[Coo98] W. Cook. *Combinatorial Optimization.* Wiley, New York, 1998.

[CPS92] R. W. Cottle, J.-S. Pang, and R. Stone. *The Linear Complementarity Problem.* Academic Press, 1992.

[DDE20] E. D. Demaine, M. L. Demaine, and D. Eppstein. Acutely Triangulated, Stacked, and Very Ununfoldable Polyhedra. In *CCCG 2020, Saskatoon, Canada, August 5–7*, 2020.

[DO07] E. D. Demaine and J. O'Rourke. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra.* Cambridge University Press, 2007.

[Dü25] A. Dürer. *The painter's manual: A manual of measurement of lines, areas, and solids by means of compass and ruler assembled by Albrecht Dürer for the use of all lovers of art with appropriate illustrations arranged to be printed in the year MDXXV.* Abaris Books, 1977 (1525).

[FEZ04] FEZ. Topexpert, 2004. `http://www.fev.com/en/what-we-do/software-and-testing-solutions/products/testing/calibration/topexpert-suite`.

[GBRS15] A. Griewank, J. U. Bernt, M. Radons, and T. Streubel. Solving piecewise linear equations in abs-normal form. *Linear Algebra and Its Applications*, 471:500–530, 2015.

[GG04] S. Gaubert and J. Gunawardena. The Perron-Frobenius Theorem for Homogeneous, Monotone Functions. *Transactions of the American Mathematical Society*, 356(12):4931–4950, 2004.

[Gho14] M. Ghomi. Affine unfoldings of convex polyhedra. *Geometry and Topology*, 18:3055–3090, 2014.

[GPFL01] K. Gschweitl, H. Pfluegl, T. Fortuna, and R. Leithgoeb. Steigerung der Effizienz in der modellbasierten Motoren-Applikation durch die neue CAMEO Online DoE-Toolbox. *ATZ – Automobiltechnische Zeitschrift*, 103(7):636–643, 2001.

[Gri13] A. Griewank. On stable piecewise linearization and generalized algorithmic differentiation. *Optimization Methods and Software*, 28(6):1139–1178, 2013.

[Grü91] B. Grünbaum. Nets of polyhedra. II. *Geombinatorics*, 1(3):5–10, 1991.

[Grü02] B. Grünbaum. No-net polyhedra. *Geombinatorics*, 11:111 – 114, 2002.

[GSL⁺18] A. Griewank, T. Streubel, L. Lehmann, M. Radons, and R. Hasenfelder. Piecewise linear secant approximation via algorithmic piecewise differentiation. *Optimization Methods and Software*, 33:1108–1126, 2018.

[Ise03] R. Isermann. *Modellgestützte Steuerung, Regelung und Diagnose von Verbrennungsmotoren.* Springer, Berlin Heidelberg, 2003.

[Ise10] R. Isermann. *Elektronisches Management motorischer Fahrzeugantriebe.* ATZ/MTZ-Fachbuch. Vieweg+Teubner, Wiesbaden, 2010.

[Ise14] R. Isermann. *Engine Modeling and Control.* Springer, Berlin Heidelberg, 2014.

[JPwP10] J. Johns, C. Painter-wakefield, and R. Parr. Linear Complementarity for Regularized Policy Evaluation and Improvement. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010.

[KKL10] T. Kruse, S. Kurz, and T. Lang. Modern statistical modeling and evolutionary optimization methods for the broad use in ECU calibration. *IFAC Proceedings Volumes*, 43(7):739–743, 2010. 6th IFAC Symposium on Advances in Automotive Control.

[KPF⁺03] K. Knödler, J. Poland, T. Fleischhauer, A. Mitterer, S. Ullmann, and A. Zell. Modellbasierte Online-Optimierung moderner Verbrennungsmotoren. *MTZ – Motortechnische Zeitschrift*, 64(6):520–526, 2003.

[M⁺16] I. Mayeres et al. Transitions towards a more sustainable mobility system. Technical Report EEA Report No. 34/2016, European Environment Agency, 2016. `https://www.eea.europa.eu/publications/term-report-2016`.

[Man07] O. L. Mangasarian. Absolute value programming. *Computational Optimization and Applications*, 36(1):43–53, 2007.

[Mat18] MathWorks. Model-based calibration toolbox, 2018. `https://de.mathworks.com/products/mbc.html`.

[MM06] O. L. Mangasarian and R. R. Meyer. Absolute value equations. *Linear Algebra and Its Applications*, 419:359–367, 2006.

[MP08] E. Miller and I. Pak. Metric Combinatorics of Convex Polyhedra: Cut Loci and Nonoverlapping Unfoldings. *Discrete & Computational Geometry*, 39(1):339–388, 2008.

[MP15] J. Merkisz and J. Pielecha. Particulate Matter Emissions during Engine Start-Up. In *Nanoparticle Emissions From Combustion Engines*, pages 47–60. Springer, Cham, 2015.

[MST18] G. P. Merker, C. Schwarz, and R. Teichmann. *Grundlagen Verbrennungsmotoren.* Springer, Berlin Heidelberg, 2018.

[MVTI03] E. Martini, H. Voß, S. Töpfer, and R. Isermann. Efficient engine calibration with local linear neural networks. *MTZ worldwide*, 64(5):19–22, 2003.

[Neu90] A. Neumaier. *Interval methods for systems of equations.* Cambridge University Press, 1990.

[Ngu07] H. Nguyen. *Gpu Gems 3.* Addison-Wesley Professional, first edition, 2007.

[O'R01] J. O'Rourke. Unfolding Prismoids without Overlap. *Unpublished manuscript*, 2001.

[O'R07] J. O'Rourke. Band Unfoldings and Prismatoids: A Counterexample. *CoRR*, abs/0710.0811, 2007.

[OR09] E. Outerelo and J. M. Ruiz. *Mapping Degree Theory.* American Mathematical Society, 2009.

[O'R12] J. O'Rourke. Unfolding Prismatoids as Convex Patches: Counterexamples and Positive Results. *CoRR*, abs/1205.2048, 2012.

[O'R17] J. O'Rourke. Addendum to: Edge-Unfolding Nearly Flat Convex Caps. *CoRR*, abs/1709.02433, 2017.

[O'R18] J. O'Rourke. Edge-Unfolding Nearly Flat Convex Caps. In Bettina Speckmann and Csaba D. Tóth, editors, *34th International Symposium on Computational Geometry (SoCG 2018)*, volume 99 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 64:1–64:14, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[OS13] G. Ottaviani and B. Sturmfels. Matrices with Eigenvectors in a Given Subspace. *Proc. Amer. Math. Soc.*, 141(4):1219–1232, 2013.

[Poi05] H. Poincaré. Sur les lignes géodésiques des surfaces convexes. *Trans. Amer. Math. Soc.*, 6:237–274, 1905.

[Rad16] M. Radons. Direct solution of piecewise linear systems. *Theoretical Computer Science*, 626:97–109, 2016.

[Rad21] M. Radons. Edge-unfolding nested prismatoids. *arXiv*, abs/2105.00555, 2021.

[Rad22] M. Radons. Edge-unfolding nested prismatoids. In *Computational Geometry: Young Researchers Forum 2022*, pages 101–104. FU Berlin, 2022.

[Roh89] J. Rohn. Systems of linear interval equations. *Linear Algebra and Its Applications*, 126:39–78, 1989.

[RTC19] M. Radons and J. Tonelli-Cueto. Generalized Perron roots and solvability of the absolute value equation. *arXiv*, abs/1912.08157, 2019.

[RTC22] M. Radons and J. Tonelli-Cueto. Generalized Perron roots and solvability of the absolute value equation. In L. F. Tabera, editor, *Discrete Mathematics Days 2022*, pages 237–242. Santander : Editorial de la Universidad de Cantabria, 2022.

[Rum97] S. M. Rump. Theorems of Perron-Frobenius type for matrices without sign restrictions. *Linear Algebra and Its Applications*, 266:1–42, 1997.

[Sch90] C. A. Schevon. *Algorithms for geodesics on convex polytopes.* PhD thesis, Johns Hopkins University, 1990.

[Sch97] W. Schlickenrieder. *Nets of polyhedra.* TU Berlin, 1997.

[Sch98] A. Schrijver. *Theory of Linear and Integer Programming.* John Wiley & Sons, 1998.

[Sch12] S. Scholtes. *Introduction to Piecewise Differentiable Equations.* Springer, 2012.

[She75] G. C. Shephard. Convex polytopes with convex nets. *Mathematical Proceedings of the Cambridge Philosophical Society*, 78(3):389–403, 1975.

[SHI00] M. Schüler, M. Hafner, and R. Isermann. Einsatz schneller neuronaler Netze zur modellbasierten Optimierung von Verbrennungsmotoren. *MTZ - Motortechnische Zeitschrift*, 61(10):704–711, 2000.

[Sie01] G. Sierksma. *Linear and Integer Programming: Theory and Practice.* CRC Press, 2nd edition, 2001.

[Ste22] E. Steinitz. IIIAB12: Polyeder und Raumeinteilungen. *Encyclopädie der mathematischen Wissenschaften*, 3:1–139, 1922.

[Tar99] A. S. Tarasov. Polyhedra that do not admitnatural unfoldings. *Uspekhi Matematicheskikh*, 1999.

[Tar08] Alexey S Tarasov. Existence of a polyhedron which does not have a non-overlapping pseudo-edge unfolding. *arXiv*, abs/0806.2360, 2008.

[Ve13] M. Vogels and et al. *DoE Model "Compression ignition engine".* AVL LIST GMBH, Graz, 2013.

[ZH22] M. Zamani and M. Hladik. Error bounds and a condition number for the absolute value equations. *Math. Program.*, 2022.

[Zie95] G. M. Ziegler. *Lectures on Polytopes.* Graduate texts in mathematics. Springer-Verlag, 1995.