Max-Planck-Institut
für Mathematik
in den Naturwissenschaften
Leipzig

Computing Convex Hulls in the Affine Building of SL_d

by

Leon Zhang

COMPUTING CONVEX HULLS IN THE AFFINE BUILDING OF SL_d

LEON ZHANG

ABSTRACT. We describe an algorithm for computing the convex hull of a finite collection of points in the affine building of $SL_d(K)$, for K a field with discrete valuation. These convex hulls describe the relations among a finite collection of invertible matrices over K. As a consequence, we bound the dimension of the tropical projective space needed to realize the convex hull as a tropical polytope.

1. INTRODUCTION

Affine buildings are infinite simplicial complexes originally introduced by Tits to study the structure of simple Lie groups. They have since found use in a variety of other contexts, including arithmetic and algebraic geometry [CHSW11, KT06], optimization [Hir18], and phylogenetics [DT98].

We consider the affine building $\mathcal{B}_d$ associated to the group $SL_d(K)$ over a discrete valued field K. There is a natural notion of convex hull in $\mathcal{B}_d$, which provides a geometric data structure for the relations among $d \times d$ invertible matrices over K. Originally introduced by Faltings [Fal01], this data structure underlies Mustafin varieties [CHSW11, HL17] and can be used to study the fundamental group of certain 3-manifolds [Sup08]. Joswig, Sturmfels, and Yu [JSY07, Algorithm 2] give a procedure for computing such a convex hull in $\mathcal{B}_d$ as the standard triangulation of a tropical convex hull in some tropical projective space. However, their algorithm requires the enumeration of all lattice points in the convex hull under consideration, which can be difficult to implement and is expensive in practice. We devise an improved algorithm with time complexity bounded in the dimension of the building and the number of matrices spanning the convex hull, making it feasible for the first time to compute convex hulls in practice.

We briefly describe the structure of this manuscript. In Section 2 we review the basics of convex lattice theory and tropical geometry that we rely on throughout. We review an algorithm for computing an apartment containing two vertices and develop its application to our problem in Section 3. We then describe our novel algorithm and prove its correctness in Section 4. In Section 5 we discuss an improvement on the previous algorithm when computing the convex hull of three lattice classes. Our algorithms have been implemented over the rational function field as a Polymake extension [GJ00]. Algorithm 5.1 has also been implemented in Mathematica over the field of rational numbers with a p-adic valuation. This software and the code for the examples in this paper can be found at our supplementary materials webpage:

$$\text{https://math.berkeley.edu/~leonyz/code/convex-hulls}$$

would also like to thank Michael Joswig and Lars Kastner for generous help on writing a Polymake extension, Petra Schwer for suggesting a geometric interpretation of Lemma 3.7, and Bernd Sturmfels for much valuable discussion and feedback.

2. Preliminaries

2.1. **Convex hulls.** We begin by fixing notation and reviewing the setup of [JSY07]. Let K be a field with discrete valuation $\mathrm{val} : K \to \mathbb{Z} \cup \{\infty\}$, let R be its valuation ring with residue field k, and π a uniformizer. Note that K^d is an R-module in a natural way.

Definition 2.2. A *lattice* Λ is an R-submodule of K^d generated by d linearly independent vectors in K^d. We often represent a lattice by an invertible matrix whose columns generate the lattice.

Let $\Lambda_1, \Lambda_2 \subseteq K^d$ be two lattices. We say that Λ_1 and Λ_2 are *equivalent* if there exists $c \in K^*$ such that $\Lambda_1 = c\Lambda_2$, and we write $[\Lambda]$ for the equivalence class of the lattice Λ. We say that two equivalence classes of lattices are *adjacent* if there exist representative lattices Λ_1 and Λ_2 respectively such that $\pi\Lambda_1 \subseteq \Lambda_2 \subseteq \Lambda_1$.

Definition 2.3. Let $\mathcal{B}_d$ be the flag simplicial complex whose 0-simplices are equivalence classes of lattices in K^d and whose 1-simplices correspond to adjacent equivalence classes. We call $\mathcal{B}_d$ the *affine building* of $SL_d(K)$.

Example 2.4. Consider the building $\mathcal{B}_2$ for $K = \mathbb{Q}_3$ the 3-adic numbers. In this case our valuation ring $R = \mathbb{Z}_3$ is the 3-adic integers and our uniformizer π is 3. The affine building $\mathcal{B}_2$ is the infinite tree with every vertex having degree 4 in Figure 1.
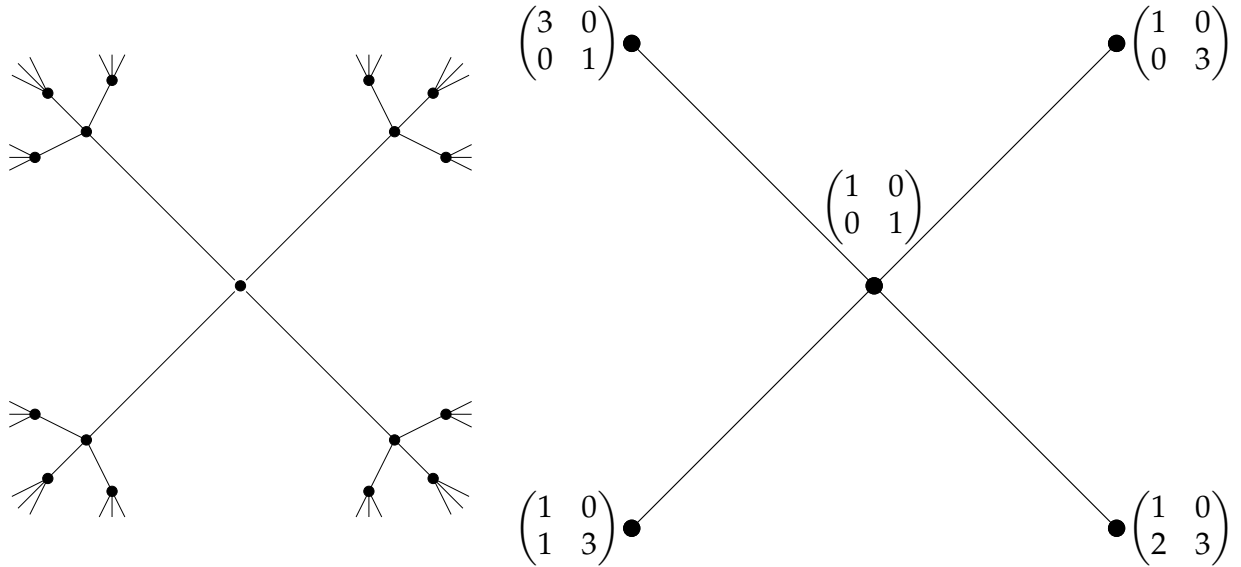

FIGURE 1. Left: the building $\mathcal{B}_2$ for $K = \mathbb{Q}_3$. Right: the link of the identity in this building.

Definition 2.5. Let $M = (v_1, \ldots, v_n)$ be a $d \times n$ matrix over K with columns $v_1, \ldots, v_n$ spanning K^d as a K-vector space, where $n > d$. The *membrane* $[M]$ of M is the collection of all lattice classes of the form $R\{\pi^{u_1}v_1, \ldots, \pi^{u_n}v_n\}$ for $u_i \in \mathbb{Z}$. If M is square, so that $d = n$, we call the membrane $[M]$ an *apartment*.

2

Lemma 2.6 ([KT06], Lemma 4.13). *Let M be a rank d matrix over K of size $d \times n$ with $n > d$. Then the membrane $[M]$ is the union of all apartments spanned by $d \times d$ invertible submatrices of M.*

Example 2.7. Let $K = \mathbb{Q}_2$, and consider the rank-2 building $\mathcal{B}_2$ over K. This is an infinite tree where every vertex has degree 3. Within $\mathcal{B}_2$, the matrix $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 2 \end{pmatrix}$ defines the membrane in Figure 2, a simplicial subtree of the building consisting of three infinite paths emanating from a vertex of degree 3.
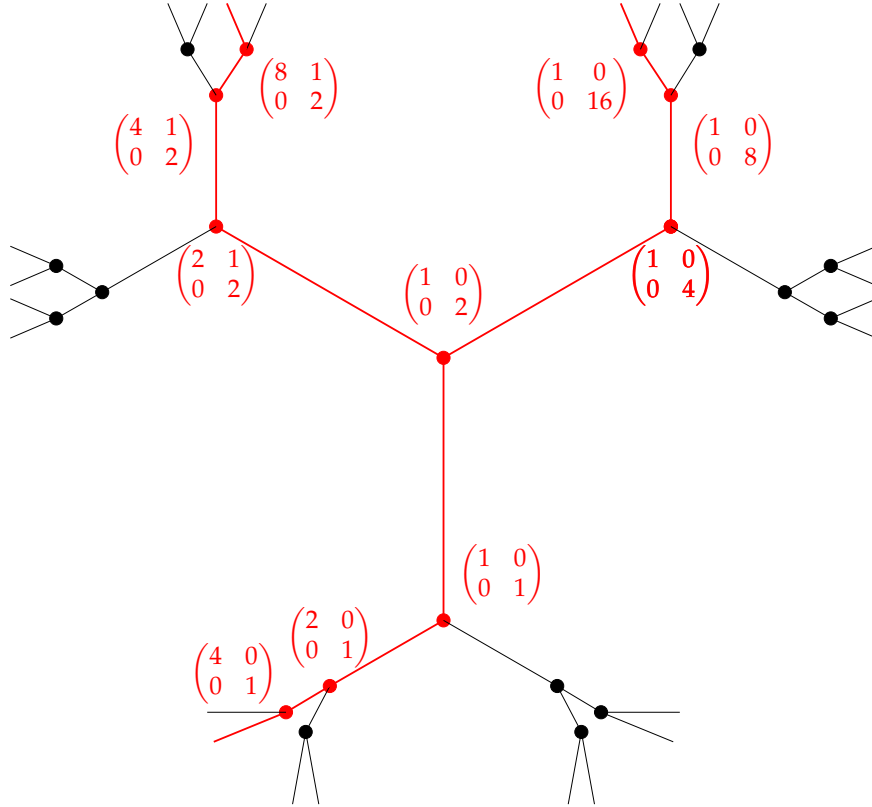

FIGURE 2. A membrane, in red, contained in the building $\mathcal{B}_2$ over $K = \mathbb{Q}_2$.

Definition 2.8. If Λ_1 and Λ_2 are lattices, then their intersection $\Lambda_1 \cap \Lambda_2$ is also a lattice. We say that a collection of lattice classes is *convex* if it is closed under taking intersections of a finite subset of representatives.

Given a finite collection of lattices $\Lambda_1, \ldots, \Lambda_s$, we call their *convex hull* $\text{conv}(\Lambda_1, \ldots, \Lambda_s)$ the smallest convex set containing their lattice classes. We can similarly define the convex hull of an infinite collection of lattices. In addition, given invertible matrices $M_1, \ldots, M_s$, we write $\text{conv}(M_1, \ldots, M_s)$ for the convex hull $\text{conv}(\Lambda_1, \ldots, \Lambda_s)$ where each Λ_i is the lattice spanned by the columns of M_i.

Remark 2.9. Our notion of convexity corresponds to *min-convexity* in the language of [JSY07]. There is another notion of convexity called *max-convexity* which arises by considering sums of lattices instead of intersections. The duality functor $\Lambda \mapsto \Lambda^* = \text{Hom}_R(\Lambda, R)$ switches sums and intersections, so via this map the max-convex hull $\text{maxconv}(\Lambda_1, \ldots, \Lambda_s)$ is isomorphic to $\text{conv}(\Lambda_1^*, \ldots, \Lambda_s^*)$. In particular, we may restrict our attention to convex hulls, and everything that follows can easily be translated to the language of max-convexity.

3

Example 2.10. Let $K = \mathbb{Q}_5$ and consider the matrices

$$M_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, M_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{5} & 0 \\ 0 & 0 & \frac{1}{125} \end{pmatrix}, M_3 = \begin{pmatrix} 5 & 625 & 150 \\ 0 & 25 & 1 \\ 0 & 0 & \frac{1}{5} \end{pmatrix}.$$

In Example 2.20 we will see that the convex hull $\mathrm{conv}(M_1, M_2, M_3)$ contains nine vertices, fifteen edges, and seven triangles, with the simplicial complex structure shown in Figure 3.
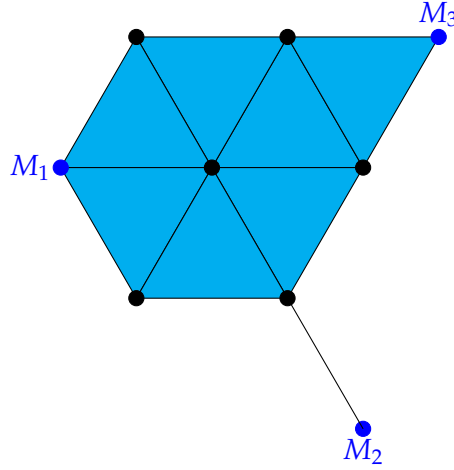

FIGURE 3. The convex hull of the matrices M_1, M_2, and M_3 in $\mathcal{B}_3$.

Lemma 2.11. *Let $\Lambda_1, \ldots, \Lambda_s$ be a finite collection of lattices. Then*

$$conv(\Lambda_1, \ldots, \Lambda_s) = \bigcup_{\Lambda' \in conv(\Lambda_2, \ldots, \Lambda_s)} conv(\Lambda_1, \Lambda').$$

Proof. Pick any class V in $\mathrm{conv}(\Lambda_1, \ldots, \Lambda_s)$ with representative $\pi^{a_1}\Lambda_1 \cap \pi^{a_2}\Lambda_2 \cap \cdots \cap \pi^{a_s}\Lambda_s$. Clearly $\Lambda' = \pi^{a_2}\Lambda_2 \cap \cdots \cap \pi^{a_s}\Lambda_s$ satisfies $[\Lambda'] \in \mathrm{conv}(\Lambda_2, \ldots, \Lambda_s)$, so $V \in \mathrm{conv}(\Lambda_1, \Lambda')$. Conversely, fix a lattice $\Lambda' = \pi^{a'_2}\Lambda_2 \cap \cdots \cap \pi^{a'_s}\Lambda_s$ representing a class in $\mathrm{conv}(\Lambda_2, \ldots, \Lambda_s)$. Any class V in $\mathrm{conv}(\Lambda_1, \Lambda')$ has a representative of the form $\pi^{a_1}\Lambda_1 \cap \pi^b\Lambda' = \pi^{a_1}\Lambda_1 \cap \pi^{a_2}\Lambda_2 \cap \cdots \cap \pi^{a_s}\Lambda_s$, where $a_i = b + a'_i$ for $i = 2, \ldots, s$. In particular, V is certainly in $\mathrm{conv}(\Lambda_1, \ldots, \Lambda_s)$. $\square$

The following result was originally stated in Faltings's paper on matrix singularities [Fal01]. For completeness we provide an easy proof.

Proposition 2.12. *Let $\Lambda_1, \ldots, \Lambda_s$ be a finite collection of lattices representing equivalence classes in $\mathcal{B}_d$. Then $conv(\Lambda_1, \ldots, \Lambda_s)$ is finite.*

Proof. Any class in $\mathrm{conv}(\Lambda_1, \Lambda_2)$ has a representative of the form $\Lambda_1 \cap \pi^a\Lambda_2$. For $a \gg 0$ we know that $\Lambda_1 \supseteq \pi^a\Lambda_2$, and for $a \ll 0$ we know $\Lambda_1 \subseteq \pi^a\Lambda_2$. Hence the convex hull of two lattices is finite. The result then follows by induction and Lemma 2.11. $\square$

It is therefore natural to ask how to compute a convex hull. In fact, the building $\mathcal{B}_d$ and membranes have an innate tropical structure which can be exploited for this purpose.

2.13. **Tropical basics.** We next review some basics of tropical convexity. For a more detailed exposition of this material, see [MS15, Chapter 4] or [Jos, Chapter 5].

We work over the tropical semiring $\mathbb{T} = (\mathbb{R} \cup \{\infty\}, \oplus, \odot)$ with the min-plus convention. In this semiring, the basic arithmetic operations of addition and multiplication are redefined:

$$a \oplus b := \min(a, b), \ a \odot b := a + b \ \text{where} \ a, b \in \mathbb{T}.$$

The *tropical projective space* $\mathbb{TP}^{n-1}$ is the space $(\mathbb{T}^n - (\infty, \infty, \ldots, \infty))/\mathbf{1}\mathbb{R}$, where $\mathbf{1} = (1, 1, \ldots, 1)$ is the all-ones vector. When illustrating this space, as in Figure 4, we always choose the affine chart in which the first coordinate is 0. There is a tropical distance metric d_{tr} in $\mathbb{TP}^{n-1}$, given by

$$d_{tr}(v, w) := \max\{|v_i - v_j + w_j - w_i| : 1 \le i < j \le n\}.$$

The lattice of integral points $\mathbb{Z}^n \subseteq \mathbb{TP}^{n-1}$ forms the skeleton of a flag simplicial complex, with a 1-simplex between two lattice points if they are of tropical distance 1 apart. This is called the *standard triangulation* of $\mathbb{TP}^{n-1}$.
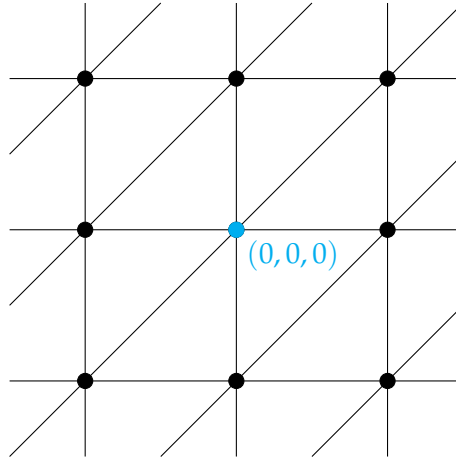

FIGURE 4. The standard triangulation of $\mathbb{TP}^2$, with the origin colored cyan.

Given a collection $\mathcal{P}$ of points in $\mathbb{TP}^{n-1}$, we define their *tropical convex hull* or *tropical polytope* as the tropical semimodule spanned by these points, i.e.:

$$\text{tconv}(\mathcal{P}) = \{\lambda_1 \odot p^{(1)} \oplus \cdots \oplus \lambda_s \odot p^{(s)} : \lambda_i \in \mathbb{T}, p^{(i)} \in \mathcal{P}\}.$$

If $\mathcal{P} \subseteq \mathbb{Z}^n$, we call their tropical convex hull a *tropical lattice polytope*.

A map p from the set of all d-sized subsets of $[n]$ to $\mathbb{T}$ satisfying the following exchange relation is called a *valuated matroid* [DT98] or *tropical Plücker vector*: for any $(d-1)$-subset σ and any $(d+1)$-subset τ of $[n]$, the minimum

$$\min\{p(\sigma \cup \{\tau_i\}) + p(\tau - \{\tau_i\}) : i \in [d+1]\}$$

is attained at least twice. (By convention we say that $p(\sigma) = \infty$ if σ has size less than d.)

A tropical Plücker vector p gives rise to a *tropical linear space* L, consisting of all points $x \in \mathbb{TP}^{n-1}$ such that, for any $(d+1)$-subset τ of $[e]$, the minimum of the numbers $p(\tau - \{\tau_i\}) + x_{\tau_i}$, for $i = 1, \ldots, d, d+1$, is attained at least twice. Given a tropical linear space L, there is a projection map pr_L taking a point $x \in \mathbb{TP}^{n-1}$ to a nearest point $pr_L(x) \in L$, which can be evaluated via the

5

red rule or *blue rule* [JSY07, Theorem 15]. We state here the blue rule, which gives a formula for the *i*th coordinate of $pr_L(x)$ in terms of an optimization over all $(d-1)$-sized subsets σ of $[n]$:

$$pr_L(x)_i = \min_\sigma \max_{j \notin \sigma}(p(\sigma \cup \{i\}) - p(\sigma \cup \{j\}) + x_j).$$

The standard triangulation of $\mathbb{TP}^{n-1}$ descends to a standard triangulation of any tropical convex hull of lattice points or of any tropical linear space L with Plücker vector image in $\mathbb{Z} \cup \{\infty\}$.

One important class of tropical linear spaces arises as follows. Let $M = (v_1, \ldots, v_n)$ be a $d \times n$ matrix of rank d over K. Then M defines a tropical Plücker vector p (and hence tropical linear space L) as follows: if ω is a collection of d integers in $[n]$, then M_ω denotes the corresponding $d \times d$ submatrix, and $p(\omega) := \mathrm{val}(\det(M_\omega))$. A tropical linear space obtained in this way is called a *tropicalized linear space*.

Example 2.14. Let M be the matrix over $\mathbb{Q}_2$ from Example 2.7: $M = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 2 \end{pmatrix}$. We can compute the Plücker vector p coming from M:

$$p(\{1,2\}) = 0, p(\{1,3\}) = 1, p(\{2,3\}) = 0.$$

The corresponding tropical linear space L consists of three rays emanating from the apex point $(0, -1, 0)$ in the $(0, 0, 1), (0, 1, 0)$, and $(1, 0, 0)$ directions of tropical projective space.
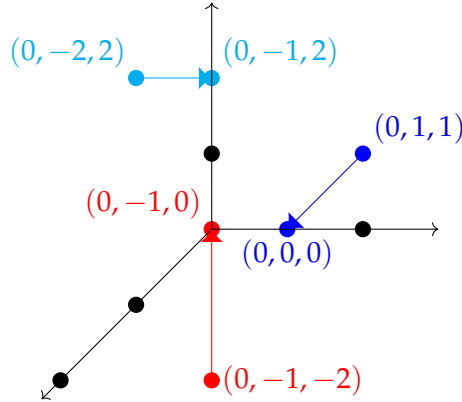

FIGURE 5.   The standard triangulation of the tropical linear space L coming from the matrix M, along with projections of three lattice points not in L.

We can now describe the tropical structure underlying the building $\mathcal{B}_d$. In effect, membranes are just standard triangulations of tropicalized linear spaces.

Theorem 2.15 ([KT06], Theorem 4.15). *Let $M = (v_1, \ldots, v_n)$ be a $d \times n$ matrix of rank d over K and let L be its associated tropical linear space. Then there is an isomorphism Ψ_M between the membrane $[M]$ and the standard triangulation of L,*

$$\Psi_M(R\{\pi^{-u_1}v_1, \ldots, \pi^{-u_n}v_n\}) := pr_L((u_1, \ldots, u_n)),$$

sending a lattice $R\{\pi^{-u_1}v_1, \ldots, \pi^{-u_n}v_n\}$ to the projection onto L of the point $(u_1, \ldots, u_n) \in \mathbb{TP}^{n-1}$.

As a first illustration of this theorem, note that Figures 2 and 5 are isomorphic as simplicial complexes. They are both trees comprising three infinite branches stemming from a single node.

Example 2.16. If our matrix M is square, so that its membrane $[M]$ is actually an apartment in the building, then Ψ_M describes a simplicial complex isomorphism between the apartment and the *tropical projective torus* $\mathbb{R}^n / \mathbb{R}\mathbf{1}$.

Example 2.17. Keep the notation of Theorem 2.15. Rincón [Rin13] describes a local structure of any tropical linear space L with Plücker vector p, in which a basis σ of the underlying matroid yields a *local tropical linear space* defined by

$$L_\sigma = \{u \in L : p(\sigma) - \sum_{i \in \sigma} u_i \le p(\tau) - \sum_{j \in \tau} u_j \ \forall \text{ bases } \tau\}.$$

These local tropical linear spaces L_σ are isomorphic to Euclidean space $\mathbb{R}^{n-1}$, are contained in L, and together form a non-disjoint cover of L.

The covering of a membrane by its apartments derives from this local structure of tropical linear spaces. In particular, let σ describe a basis of the matroid of M, so that the $d \times d$ matrix M_σ with columns indexed by σ is invertible. Then the linearity of the determinant over column sums implies that the apartment $[M_\sigma]$ is mapped by Ψ_M to the local tropical linear space L_σ.

Given any membrane $[M]$ represented by a $d \times n$ matrix $M = (f_1, \ldots, f_n)$, there is a *retraction* r_M of the entire building $\mathcal{B}_d$ onto $[M]$, which restricts to the identity on $[M]$ itself:

$$r_M : \Lambda \mapsto (\Lambda \cap K\{f_1\}) + \cdots + (\Lambda \cap K\{f_n\}).$$

We may use this map to describe a tropical structure for convex hulls.

Theorem 2.18 ([JSY07], Proposition 22). *Let M be a $d \times n$ matrix of rank d over K, $[M]$ its corresponding membrane, and L its corresponding tropical linear space. Also let $\Lambda_1, \ldots, \Lambda_s$ be lattices corresponding to points in $\mathcal{B}_d$. The following two simplicial complexes coincide:*

$$r_M(conv(\Lambda_1, \ldots, \Lambda_s)) \subseteq [M],$$
$$tconv(\Psi_M(r_M(\Lambda_1)), \ldots, \Psi_M(r_M(\Lambda_s))) \subseteq L.$$

In particular, if $[M]$ contains the convex hull of $\Lambda_1, \ldots, \Lambda_s$, then the retraction map acts as the identity, and the convex hull $conv(\Lambda_1, \ldots, \Lambda_s)$ is isomorphic to the standard triangulation of a tropical polytope. This suggests an approach for computing convex hulls in $\mathcal{B}_d$ as follows:

Algorithm 2.19 (Convex hull computation).
INPUT: $M_1, \ldots, M_s$ $d \times d$ invertible matrices over K whose columns are bases for lattices $\Lambda_1, \ldots, \Lambda_s$
OUTPUT: $conv(\Lambda_1, \ldots, \Lambda_s) \subseteq \mathcal{B}_d$
 1: $[M] \leftarrow$ a membrane containing $conv(\Lambda_1, \ldots, \Lambda_s)$
 2: **for all** $i \in \{1, \ldots, s\}$ **do**
 3: $\quad P_i \leftarrow \Psi_M(r_M(\Lambda_i))$
 4: **end for**
 5: $X \leftarrow tconv(P_1, \ldots, P_s)$
 6: **return** X

Note that given a lattice Λ_i represented by a matrix M_i, we can compute the image $\Psi_M(r_M(\Lambda_i))$ in $\mathbb{TP}^{n-1}$ simply by taking the tropical row sum of the matrix $val(M_i^{-1}M)$, by [JSY07, Lemma 21].

Example 2.20. Retain the setup of Example 2.10. We can apply Algorithm 5.1, which we will soon discuss, to get that the membrane $[M]$ with

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 5 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

contains the convex hull of M_1, M_2, and M_3. Running through Algorithm 2.19 with this membrane yields the following tropical matrix,

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 \\ -1 & -2 & 1 & -1 \end{pmatrix},$$

whose rows or columns span the tropical convex hull in Figure 3. Note that tropical polytopes are self-dual, i.e. the columns and rows of any tropical matrix span isomorphic tropical polytopes [DS04, Theorem 1].
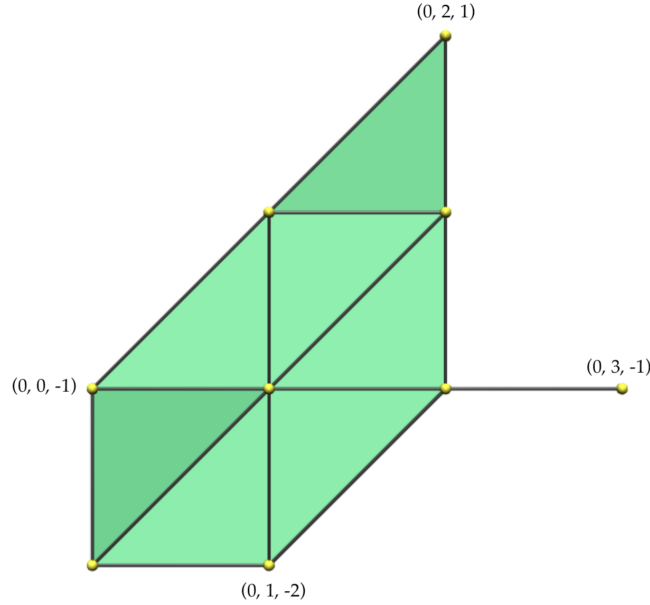

FIGURE 6. The tropical polytope isomorphic to the convex hull of the three matrices in Example 2.10, with coordinates of spanning vertices labeled.

From this tropical polytope we can construct representatives for any of the lattice classes in Figure 3. For example, consider the central lattice point $(0, 1, -1)$ with six neighbors. By Equation (14) in [DS04], this lattice point in the column-span of our tropical matrix corresponds to $(-1, -1, 0, 0)$ in the row-span. In turn, this point corresponds to the class of the lattice

$$\mathbb{Z}_5 \left\{ 5^1 \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, 5^1 \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, 5^0 \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, 5^0 \cdot \begin{pmatrix} 0 \\ 5 \\ 1 \end{pmatrix} \right\}.$$

Of course, Algorithm 2.19 requires an *enveloping membrane* of $\Lambda_1, \ldots, \Lambda_s$: a membrane containing the convex hull of $\Lambda_1, \ldots, \Lambda_s$. Without such a membrane the tropical polytope produced by Algorithm 2.19 need not be isomorphic to our original convex hull. Note that membranes need not be convex, so it is not sufficient to simply find a membrane containing the spanning lattice classes. This fact will be demonstrated in Example 4.8.

No bounded-time algorithm for computing an arbitrary enveloping membrane has thus far been described. A procedure for computing such an enveloping membrane is described in [JSY07], but it is often impractical: it relies on the computation of each individual element of the convex hull,

8

while expanding a starting membrane to contain each element whenever necessary. In Section 4 we will describe an improved algorithm with bounded complexity in d and s, allowing us to algorithmically realize any convex hull as a tropical polytope.

Remark 2.21. Our notion of convexity was originally introduced by Faltings [Fal01], who noted that configurations Γ of vertices in $\mathcal{B}_d$ correspond to certain schemes $M(\Gamma)$ called *Mustafin varieties* or *Deligne schemes* over the spectrum of a DVR. In the arithmetic setting, these Mustafin varieties function as local models of Shimura varieties [PRS13]. The special fiber of $M(\Gamma)$ generally has many singularities, but replacing Γ with its convex hull Γ' yields a regular Mustafin variety $M(\Gamma')$ with a dominant morphism $M(\Gamma') \to M(\Gamma)$, such that the irreducible components of the special fiber of $M(\Gamma')$ intersect transversally [CHSW11, Lemma 2.4 and Theorem 2.10]. In this way, our fully-specified version of Algorithm 2.19, derived in Section 4, allows for the explicit resolution of singularities of Mustafin varieties.

3. Simultaneously-adaptable bases

In this section we review a classical result on lattices over valued fields, following [AB08, Section 6.9], and describe its relevance to our setting of convex hulls in affine buildings. Note that in what follows we say *monomial matrix* to refer to any matrix A supported on a permutation matrix: i.e., there exists a permutation σ such that $A_{ij} \neq 0$ if and only if $j = \sigma(i)$.

Algorithm 3.1 (Simultaneously adaptable basis for two lattices).
INPUT: M_1, M_2 $d \times d$ invertible matrices over K whose columns are bases for lattices Λ_1 and Λ_2 in $\mathcal{B}_d$
OUTPUT: Invertible matrix A, monomial matrix Δ such that the columns of A and $A\Delta$ are bases for Λ_1 and Λ_2, respectively
1: $B_1 \leftarrow M_1$
2: $C_1 \leftarrow M_2$
3: **for all** $i \in \{1, \ldots, n-1\}$ **do**
4:     $N_i \leftarrow B_i^{-1} C_i$
5:     $n_i \leftarrow$ entry of minimal valuation in N_i not equal to $n_1, \ldots, n_{i-1}$
6:     $L_i \leftarrow$ the matrix such that $L_i N_i$ is obtained from N_i by eliminating all other elements in the column of n_i
7:     $R_i \leftarrow$ the matrix such that $L_i N_i R_i$ is obtained from $L_i N_i$ by eliminating all other elements from the row of n_i
8:     $B_{i+1} \leftarrow B_i L_i^{-1}$
9:     $C_{i+1} \leftarrow C_i R_i$
10: **end for**
11: $A \leftarrow B_d$
12: $\Delta \leftarrow B_d^{-1} C_d$
13: **return** A, Δ

Lemma 3.2. *Let M_1 and M_2 be $d \times d$ invertible matrices over K for lattices Λ_1 and Λ_2 in $\mathcal{B}_d$. Then Algorithm 3.1 correctly returns a basis A for Λ_1 and a monomial matrix Δ such that $A\Delta$ is a basis for Λ_2.*

Proof. Because n_i is chosen to be of minimal valuation in N_i, each L_i and R_i will be matrices in $SL_d(R)$. It follows that the new matrices B_{i+1} and C_{i+1} will be bases for Λ_1 and Λ_2 if B_i and C_i are, with base change matrix $L_i N_i R_i$. In particular, after $d-1$ steps of the for-loop, the matrix $\Delta = L_{d-1} N_{d-1} R_{d-1}$ will have $d-1$ distinct entries which are uniquely nonzero in their respective rows and columns. Hence Δ is a monomial matrix, as desired. $\square$

Definition 3.3. We call the output A in Algorithm 3.1 a *simultaneously adaptable basis (SA-basis)* for Λ_1 and Λ_2.

Algorithm 3.1 is a demonstration of the building-theoretic fact that any two lattices lie in a common apartment. In general, there should be many distinct apartments containing any two given points. Indeed, the SA-basis A obtained above depends not only on the lattices Λ_1 and Λ_2 but on our original choice of bases, which lattice we designate as Λ_1, and how we break "ties" between elements of minimal valuation when choosing pivots. In what follows, we break ties between potential pivots by picking the option in the leftmost column, then topmost row.

Because apartments are min-convex, we have the following fact:

Corollary 3.4. *Pick two lattice classes represented by Λ_1 and Λ_2 in $\mathcal{B}_d$, and let A be a SA-basis for the two lattices. Then the apartment $[A]$ contains the convex hull $conv(\Lambda_1, \Lambda_2)$.*

We can therefore view Algorithm 3.1 as a procedure for computing an apartment containing the convex hull of two points.

Lemma 3.5. *Let M_1 and M_2 be two invertible $d \times d$ matrices representing lattices Λ_1 and Λ_2, and let Γ be any diagonal matrix with $M_2' = M_2\Gamma$. Let N_i and N_i' be the base-change matrices at the ith step of Algorithm 3.1 executed with the pairs (M_1, M_2) and (M_1, M_2') as input respectively, n_i and n_i' the chosen pivots of least valuation at step i, and so on. If k is a positive integer such that the positions of the pivots n_i and n_i' agree for all i up to $k-1$, then $L_{k-1} = L_{k-1}'$ and $N_k' = N_k\Gamma$.*

Proof. We prove the result by induction, noting that the base case $k = 1$ follows trivially. Suppose that the first $k-1$ pivots are the same for the two algorithm executions. Because the first $k-2$ pivots are the same, by the inductive hypothesis we have that $N_{k-1}' = N_{k-1}\Gamma$. In particular, the ratio of any two entries in the same column is the same for N_{k-1} and N_{k-1}'. Now since the $k-1$st pivot position is also the same, the row operations to obtain N_k and N_k' from N_{k-1} and $N_{k-1}' = N_{k-1}\Gamma$ agree as well, so that $L_{k-1} = L_{k-1}'$. Next the column operations necessary to clear the rows of two pivots may differ, but in both executions we eliminate using a column which has no other nonzero entries. It follows that $N_k' = N_k\Gamma$, as desired. $\square$

Corollary 3.6. *Keep the setup of Lemma 3.5 above, and let A be the basis of Λ_1 produced by Algorithm 3.1. If all pivot positions of Algorithm 3.1 are the same for the two inputs (M_1, M_2) and (M_1, M_2'), then the lattice class $[\Lambda_2']$ for M_2' is contained in the apartment $[A]$.*

Proof. Because all pivots are the same, Lemma 3.5 implies that $L_i = L_i'$ for all i. This means the final basis for Λ_1 produced by both executions of the algorithm is $A = L_d L_{d-1} \ldots L_1 M_1$. In particular, we have that $[A]$ contains both $[\Lambda_2]$ and $[\Lambda_2']$. $\square$

Lemma 3.7. *Let A be an invertible matrix defining an apartment $[A]$ and M a basis for a lattice Λ whose class is not in $[A]$. Then $conv([\Lambda], [A])$ can be covered with $d!$ different apartments.*

Proof. Any element of $conv([\Lambda], [A])$ will be contained in some $conv([\Lambda], [\Lambda'])$ where Λ' is a lattice having basis $A\Gamma$ for some diagonal matrix Γ. Fix such a Γ. We can use Algorithm 3.1 to compute an apartment $[B]$ containing $conv([\Lambda], [A\Gamma])$. By Corollary 3.6, this apartment also contains $conv([\Lambda], [A\Gamma'])$ for any other diagonal matrix Γ' leading to the same sequence of pivot positions. The key point is that we need only consider the sequence of columns that the pivots appear in. Since each pivot must appear in a different column, this means there are $d!$ different sequences of pivot positions.

To see why only the sequence of columns of the pivots matters, take some other Γ' diagonal, and suppose that the sequence of pivot columns is the same for the two executions $(M, A\Gamma)$ and $(M, A\Gamma')$. We prove by induction on the jth pivot that all pivots are actually in the same positions. The first $j - 1$ pivots appear in the same positions by assumption, so by Lemma 3.5 the jth base-change matrix N'_j for the input $(M, A\Gamma')$ equals $N_j\Gamma^{-1}\Gamma'$, where N_j is the jth base-change matrix for the input $(M, A\Gamma)$. Then since the jth pivots appear in the same column, and scaling columns does not change the column entry of minimal valuation, the jth pivot will be in the same position for both executions as well. □

Remark 3.8. We note the similarity of Lemma 3.7 with [Hit11, Lemma 6.3], which states that any apartment A in any building can be covered by the union of Weyl chambers based at some other fixed point z with equivalence class in ∂A, the spherical apartment at infinity corresponding to A. We expect that Lemma 3.7 is an explicit analogue of this result in our specialized setting, where ∂A is isomorphic to the symmetric group S_d on d elements, in which each Weyl chamber is replaced by a suitable apartment containing it to ensure the the convex hull of z and A is also covered.

4. Constructing enveloping membranes

In this section we combine the results of the previous section to solve the problem left open in Algorithm 2.19. Namely, we present an algorithm to compute an enveloping membrane of a finite set of lattices. This allows us to realize convex hulls in the building as tropical polytopes.

Algorithm 4.1 (List of apartments covering a convex hull).
INPUT: $B_1, \ldots, B_s$ base matrices for lattices $\Lambda_1, \ldots, \Lambda_s$
OUTPUT: A set of apartments covering $\mathrm{conv}(\Lambda_1, \ldots, \Lambda_s)$
  1: **if** $s = 2$ **then**
  2:    **return** SA-basis of Λ_1 and Λ_2 via Algorithm 3.1
  3: **end if**
  4: $L_{s-1} \leftarrow$ set of apartments covering $\mathrm{conv}(\Lambda_2, \ldots, \Lambda_s)$ via Algorithm 4.1
  5: $L_s \leftarrow \varnothing$
  6: **for all** $[A] \in L_{s-1}$ **do**
  7:    $L_A \leftarrow$ set of apartments covering $\mathrm{conv}(\Lambda_1, [A])$ as in Remark 4.3
  8:    $L_s \leftarrow L_s \cup L_A$
  9: **end for**
10: **return** L_s

Theorem 4.2. *Let $M_1, \ldots, M_s$ represent s lattices $\Lambda_1, \ldots, \Lambda_s$ in $\mathcal{B}_d$. Then Algorithm 4.1 correctly computes a list of apartments L_s such that each lattice class $[\Lambda] \in \mathrm{conv}(\Lambda_1, \ldots, \Lambda_s)$ is contained in $[A]$ for some $[A] \in L_s$. Furthermore, L_s has size at most $(d!)^{s-2}$.*

Of course this theorem and Lemma 2.6 together imply that Algorithm 4.1 can be used to compute an enveloping membrane for $\Lambda_1, \ldots, \Lambda_s$. We simply concatenate all the matrices in the output L_s.

Proof. If the algorithm is correct, then L_{s-1} contains at most $(d!)^{s-3}$ apartments by induction. Since L_A has size at most $d!$ by Lemma 3.7, L_s has size at most $(d!)^{s-2}$.

It remains to prove correctness. By Lemma 2.11, any lattice class $[\Lambda]$ in $\mathrm{conv}(\Lambda_1, \ldots, \Lambda_s)$ is contained in $\mathrm{conv}(\Lambda_1, \Lambda')$ for some $[\Lambda'] \in \mathrm{conv}(\Lambda_2, \ldots, \Lambda_s)$. There exists some $[A] \in L_{s-1}$ such that $[\Lambda'] \in [A]$, and so $[\Lambda] \in \mathrm{conv}(\Lambda_1, [A])$. In particular, there is some $[B] \in L_A$ such that $[\Lambda] \in [B]$. □

Remark 4.3. The crucial part of Algorithm 4.1 is computing the set L_A of apartments covering $conv(\Lambda_1, [A])$. Recall from Lemma 3.7 that this set is indexed by permutations in S_d. We sketch here how to compute the apartment corresponding to the identity permutation; all other apartments can be computed very similarly.

Let $\Gamma = diag(\pi^{a_1}, \ldots, \pi^{a_d})$ be a diagonal matrix with indeterminates $a_1, \ldots, a_d \in \mathbb{Z}$. First choose Γ to be any diagonal matrix such that the first pivot for the first base change matrix $M_1^{-1} A\Gamma$ is in the first column. Next we compute the second base-change matrix; by decreasing both a_1 and a_2 by a large enough common value, Lemma 3.5 guarantees that the first pivot will still be in the first column, and that the second pivot will appear in the second column. We next compute the third base-change matrix by reducing a_1, a_2, and a_3 all by some large enough value, and so on.

Remark 4.4. We present in the next section a more efficient algorithm for the $s = 3$ case, Algorithm 5.1, needing only 2^d apartments to cover the convex hull instead of $d!$. Because Algorithm 4.1 is inductive on s, Algorithm 5.1 can be used for the $s = 3$ case, providing a slightly better overall bound of $2^d \cdot (d!)^{s-3}$ apartments needed to cover the convex hull of s lattices.

Corollary 4.5. *Let $\Lambda_1, \ldots, \Lambda_s$ be lattices in K^d. Then their convex hull $conv(\Lambda_1, \ldots, \Lambda_s)$ is isomorphic to a tropical polytope in $\mathbb{TP}^N$ where $N \leq d \cdot 2^d \cdot (d!)^{s-3} - 1$.*

Proof. Remark 4.4 implies that a matrix M with at most $d \cdot 2^d \cdot (d!)^{s-3}$ columns generates a membrane $[M]$ containing $conv(\Lambda_1, \ldots, \Lambda_s)$. Algorithm 2.19 then realizes the convex hull as a tropical polytope in a tropical projective space of dimension at most $d \cdot 2^d \cdot (d!)^{s-3} - 1$. $\square$

Corollary 4.6. *Let $\Lambda_1, \ldots, \Lambda_s$ be lattices in K^d. Then their convex hull $conv(\Lambda_1, \ldots, \Lambda_s)$ is isomorphic to a tropical polytope spanned by at most $d \cdot 2^d \cdot (d!)^{s-3}$ points in $\mathbb{TP}^{s-1}$.*

Proof. This follows directly from Corollary 4.5 and the self-duality of tropical polytopes. $\square$

Corollary 4.7. *Let $\Lambda_1, \ldots, \Lambda_s$ be lattices in K^d. Let $[M]$ be the enveloping membrane for $conv(\Lambda_1, \ldots, \Lambda_s)$ computed by concatenating the apartments from Algorithm 4.1. Then Λ_1 is mapped to the origin by Ψ_M in Algorithm 2.19.*

Proof. There is another representation of the building $\mathcal{B}_d$, which describes the vertices as additive norms $N : K^d \to \mathbb{R} \cup \{\infty\}$. We can easily pass between these two descriptions of the building in terms of lattice classes and additive norms. If Λ is a lattice represented by a matrix M, then the corresponding additive norm is defined by

$$N_\Lambda(v) = \max\{u \in \mathbb{Z} : z^{-u}v \in \Lambda\}.$$

Write $M = (v_1, \ldots, v_n)$. By [JSY07, Lemma 21], the image of Λ_1 under the map of Theorem 2.18 is $(N_{\Lambda_1}(v_1), \ldots, N_{\Lambda_1}(v_n))$, where N_{Λ_1} is the additive norm corresponding to Λ_1. But clearly $N_{\Lambda_1}(v_i) = 0$ for each i, since each v_i is an element for a basis for Λ_1. $\square$

Viewed in the dual setting of Corollary 4.6, Corollary 4.7 implies that our algorithm places us in the affine chart of $\mathbb{TP}^{s-1}$ where the first coordinate is zero.

Example 4.8. Consider the following four 3×3 matrices over $\mathbb{C}((t))$:

$$M_1 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & t & t^2 \\ 1 & t^{-2} & t \end{pmatrix}, M_2 = \begin{pmatrix} 1 & 1 & 1 \\ t & t^2 & t^3 \\ t^{-2} & t & t^5 \end{pmatrix}, M_3 = \begin{pmatrix} 1 & 1 & 1 \\ t^2 & t^3 & t^4 \\ t & t^5 & t^8 \end{pmatrix}, M_4 = \begin{pmatrix} 1 & 1 & 1 \\ t^3 & t^4 & t^5 \\ t^5 & t^8 & t^{12} \end{pmatrix}.$$

These are the contiguous maximal submatrices of

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & t & t^2 & t^3 & t^4 & t^5 \\ 1 & t^{-2} & t & t^5 & t^8 & t^{12} \end{pmatrix},$$

so the corresponding lattice classes certainly all lie in the membrane $[M]$. An optimist could suppose that $[M]$ were in fact an enveloping membrane for the convex hull of our four matrices. Running through Algorithm 2.19 with the membrane $[M]$ yields the following tropical matrix:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 & 0 \\ -3 & -4 & 0 & 0 & 0 & 0 \\ -6 & -8 & -5 & 0 & 0 & 0 \end{pmatrix}.$$

The columns of this matrix span the tropical polytope P, visualized using Polymake in Figure 7. Its standard triangulation contains 18 vertices, 32 edges, and 15 triangles.
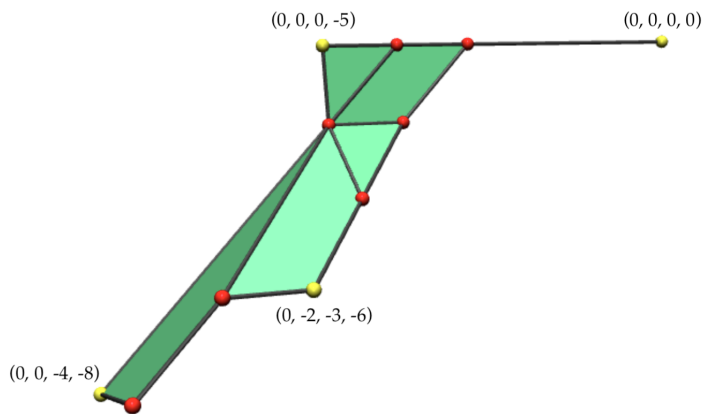

FIGURE 7. The tropical polytope P obtained by using the membrane $[M]$ for the lattices $M_1, M_2, M_3,$ and M_4 with Algorithm 2.19. Points spanning the tropical convex hull are marked in yellow.
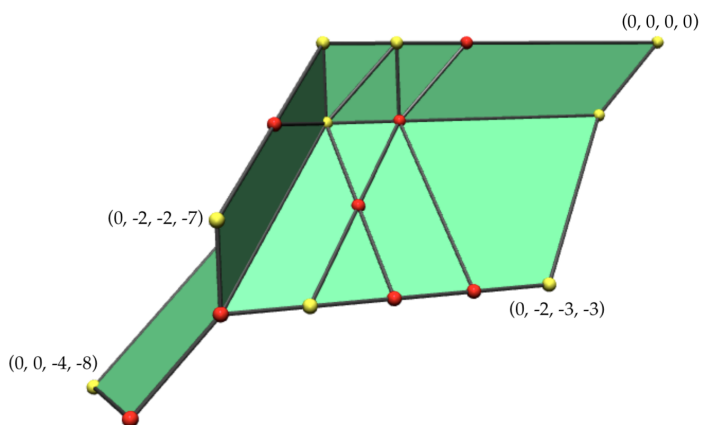

FIGURE 8. The tropical polytope P' whose standard triangulation is isomorphic to the convex hull of $M_1, M_2, M_3,$ and M_4, with spanning vertices marked in yellow.

However, when we run Algorithm 4.1 in Polymake to compute an enveloping membrane for $\mathrm{conv}(M_1, M_2, M_3, M_4)$, we obtain a different matrix M' with 12 distinct columns. Executing Algorithm 2.19 using the membrane $[M']$ yields that $\mathrm{conv}(M_1, M_2, M_3, M_4)$ is isomorphic as a simplicial complex to the tropical polytope P' in Figure 8 spanned by

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & -2 & 0 & -2 & 0 & 0 & -2 & 0 & 0 \\ -3 & -4 & 0 & -1 & -2 & 0 & -2 & 0 & -1 & -3 & -1 & 0 \\ -6 & -8 & -5 & -5 & -7 & 0 & -7 & -4 & -1 & -3 & -5 & 0 \end{pmatrix}.$$

The standard triangulation of this polytope contains 29 lattice points, 67 edges, and 41 triangles. In particular, the convex hull of $M_1, M_2, M_3,$ and M_4 is larger than the polytope P obtained via the

membrane $[M]$, even though each lattice spanning the convex hull is trivially contained in $[M]$. In turn, this means that $[M]$ does not contain the convex hull $\mathrm{conv}(M_1, M_2, M_3, M_4)$, demonstrating the fact that membranes are not convex.

Example 4.9. Let $K = \mathbb{C}((t))$ be the field of formal complex Laurent series, and let M_1 be the 4×4 identity matrix, M_2 be diagonal with entries $1, t^3, t^{-2}$, and t^{-2} respectively, and

$$M_3 = \begin{pmatrix} t^{-3} - t^2 & 1 - t^2 & -t^{-2} + 1 & t^{-2} - t \\ t^2 - t^3 & -t^{-3} + t & 1 - t & 0 \\ 0 & -1 + t & t^{-3} - t^3 & t^{-3} - 1 \\ -t + t^2 & -t^{-1} + 1 & 0 & -t^{-1} + t^2 \end{pmatrix},$$

$$M_4 = \begin{pmatrix} 1 - t^3 & t^{-1} - 1 & 1 - t^2 & 1 - t^3 \\ t^{-3} - 1 & 1 - t & 1 - t^2 & 1 - t^3 \\ -t^{-3} + t & -t^{-2} + 1 & -t^{-3} + t^{-1} & -1 + t \\ t^{-3} - t^{-2} & -t^{-1} + 1 & -t^{-1} + 1 & t^{-1} - 1 \end{pmatrix}.$$

Concatenating the matrices produced by Algorithm 4.1 applied to M_1, M_2, M_3, and M_4 in Polymake gives a matrix M with 84 distinct columns. Using the corresponding membrane $[M]$ with Algorithm 2.19, we get a 4×84 matrix over the tropical numbers. After pruning duplicate columns, we obtain the following matrix whose tropical row or column span gives the polytope displayed in Figure 9. The triangulation of that polytope has 30 vertices, 95 edges, 102 triangles, and 36 tetrahedra.

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -3 & -2 & -1 & -3 & -3 & 0 & 2 & 2 & 0 & -1 & -1 & -3 \\ 1 & 2 & 3 & 1 & 1 & 3 & 3 & 1 & 1 & 1 & 3 & 3 \\ 3 & 3 & 3 & 2 & 1 & 1 & 3 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$
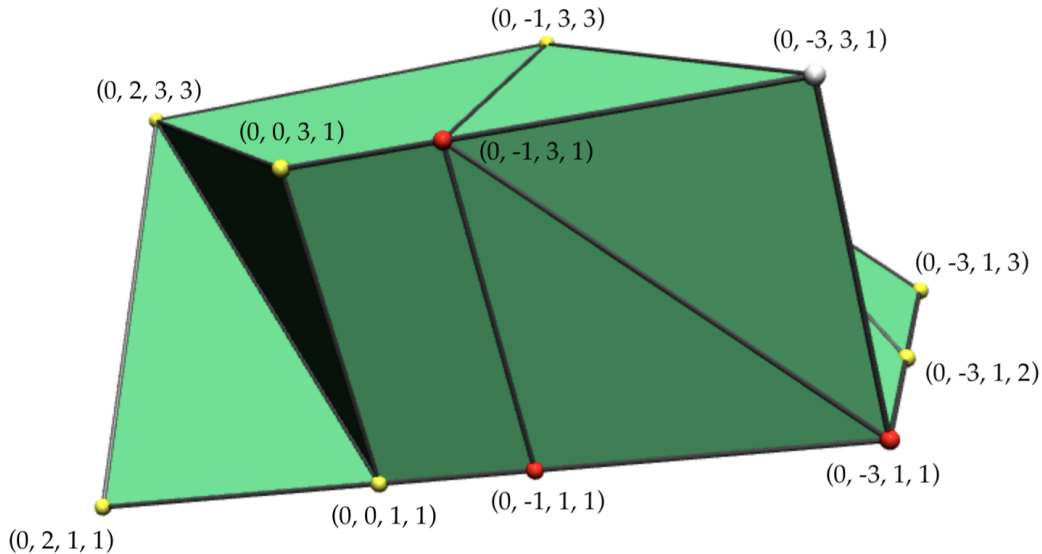

FIGURE 9. The 3-dimensional tropical polytope isomorphic to the convex hull of our matrices M_1, M_2, M_3, M_4, whose standard triangulation has f-vector (30, 95, 102, 36).

5. CONVEX TRIANGLES

Suppose that $s = 3$, so that we wish to compute a *convex triangle*: the convex hull of three lattice classes. This is relevant e.g. to [CHSW11, Section 4.6], which focuses on Mustafin varieties arising from convex triangles. In this case there exists a more efficient algorithm, taking advantage of the fact that $\mathrm{conv}(\Lambda_2, \Lambda_3)$ is just a path in the building. We now describe this improvement.

With some extra book-keeping, note that Algorithm 3.1 can output all of the following:

- an SA-basis A which is a basis for Λ_1,
- a diagonal matrix $\Delta = \mathrm{diag}(\pi^{c_1}, \ldots, \pi^{c_d})$ such that $A\Delta$ is a basis for Λ_2, where $c_1 \leq \cdots \leq c_d$,
- all of the base change matrices $N_1, \ldots, N_d$,
- and the positions $p_1, \ldots, p_d$ of the pivots $n_1, \ldots, n_d$.

We justify the existence of such a Δ. First, note that the base-change matrix Δ produced by Algorithm 3.1 can always be taken to be diagonal, since other monomial matrices correspond simply to reordering the scaled basis vectors of Λ_1. Second, we may reorder the columns of A itself in any way we like; in particular, we can order them so that the matrix Δ has the structure described above.

Algorithm 5.1 (Enveloping membrane for a convex triangle).
INPUT: M_1, M_2, M_3 three $d \times d$ invertible matrices over K whose columns are bases for lattices $\Lambda_1, \Lambda_2, \Lambda_3$ in $\mathcal{B}_d$
OUTPUT: A list L of apartments covering $\mathrm{conv}(\Lambda_1, \Lambda_2, \Lambda_3)$.
1: $L \leftarrow \varnothing$
2: $(A, \Delta = \mathrm{diag}(\pi^{c_1}, \ldots, \pi^{c_d})) \leftarrow d \times d$ matrices such that A is a basis for Λ_2 and $A\Delta$ is a basis for Λ_3, with $c_1 \leq c_2 \leq \ldots \leq c_d$
3: **for all** $i \in \{1, \ldots, d-1\}$ **do**
4: $\quad \lambda \leftarrow c_i$
5: $\quad \Gamma_\lambda \leftarrow \mathrm{diag}(\pi^{\max(\lambda, c_1)}, \ldots, \pi^{\max(\lambda, c_d)})$
6: $\quad t \leftarrow 0$
7: $\quad$ **while** $\lambda < c_{i+1}$ **do**
8: $\quad\quad \lambda \leftarrow \lambda + t$
9: $\quad\quad A_\lambda \leftarrow$ an SA-basis for $(M_1, A\Gamma_\lambda)$
10: $\quad\quad (N_1, \ldots, N_d) \leftarrow$ the sequence of base-change matrices in the SA-basis computation for $(M_1, A\Gamma_\lambda)$
11: $\quad\quad (p_1, \ldots, p_d) \leftarrow$ the sequence of pivot positions in the SA-basis computation for $(M_1, A\Gamma_\lambda)$

12: $\quad\quad L \leftarrow L \cup \{[A_\lambda]\}$
13: $\quad\quad t \leftarrow c_{i+1} - c_i$
14: $\quad\quad$ **for all** $j \in \{1, \ldots, d\}$ such that p_j is in the first i columns **do**
15: $\quad\quad\quad v_1 \leftarrow$ valuation of jth pivot in N_j
16: $\quad\quad\quad v_2 \leftarrow$ least valuation among all elements of N_j in columns $i+1, i+2, \ldots, d$ not in positions $p_1, \ldots, p_j$
17: $\quad\quad\quad t \leftarrow \min(t, v_2 - v_1 + 1)$
18: $\quad\quad$ **end for**
19: $\quad$ **end while**
20: **end for**
21: **return** L

Theorem 5.2. *Let $M_1, M_2,$ and M_3 represent three lattices $\Lambda_1, \Lambda_2,$ and Λ_3 in $\mathcal{B}_d$. Then Algorithm 5.1 correctly computes a list L of apartments covering $conv(\Lambda_1, \Lambda_2, \Lambda_3)$, where L has size at most 2^d.*

As before, we can obtain an enveloping membrane for $\Lambda_1, \Lambda_2,$ and Λ_3 by concatenating all matrices in L.

Proof. In the setup of Algorithm 5.1, any class in $conv(\Lambda_2, \Lambda_3)$ has a representative of the form $A\Gamma_\lambda$, where $\Gamma_\lambda = diag(\pi^{\max(\lambda, c_1)}, \dots, \pi^{\max(\lambda, c_d)})$ and λ is an integer between c_1 and c_d. It follows from Lemma 2.11 that

$$conv(\Lambda_1, \Lambda_2, \Lambda_3) = \bigcup_{c_1 \le \lambda \le c_d} conv(M_1, A\Gamma_\lambda).$$

We can therefore cover $conv(\Lambda_1, \Lambda_2, \Lambda_3)$ with the apartments $[A_\lambda]$ containing $conv(M_1, A\Gamma_\lambda)$ produced by Algorithm 3.1. By Corollary 3.6, furthermore, if we have computed A_λ already we only need to compute $A_{\lambda+1}$ if some pivot changes position.

Suppose this occurs, with λ in the range $c_i \le \lambda < c_{i+1}$. Then $B\Gamma_{\lambda+1}$ is obtained from $A\Gamma_\lambda$ by multiplying with the diagonal matrix whose first i diagonal entries are π and last $d - i$ diagonal entries are 1. Let p_j be the earliest pivot which changes positions. By Lemma 3.5, it follows that the jth base-change matrix $N_j^{(\lambda+1)}$ for the pair $(M, A\Gamma^{\lambda+1})$ factors as $N_j^{(\lambda+1)} = N_j^{(\lambda)} diag(\pi, \dots, \pi, 1, \dots, 1)$, where $N_j^{(\lambda)}$ is the jth base-change matrix for the pair $(M, A\Gamma^\lambda)$. Since the jth pivot differs for these two matrices, the jth pivot must appear in the first i columns and there must be an element of equal valuation appearing in the last $d - i$ columns. Conversely, suppose there exists some jth pivot appearing in the first i columns of $N_j^{(\lambda)}$ with an element of equal valuation in the last $d - i$ columns. Then either some earlier pivot already changed, or the jth pivot will be different for $N_j^{(\lambda+1)}$.

It follows that, for λ in the range $c_i \le \lambda < c_{i+1}$, we can quickly compute the smallest t such that $(M_1, A\Gamma_{\lambda+t})$ will have some jth pivot in a different position than for $(M_1, A\Gamma_\lambda)$. For each jth pivot appearing in the first i columns of $N_j^{(\lambda)}$, we can compare its valuation v_1 to the smallest valuation v_2 of all elements in the last $d - i$ columns of $N_j^{(\lambda)}$. If p_j is the first pivot to change, it will change when $t = t_j := v_2 - v_1 + 1$. So $t = \min(t_j)$ is our desired increment. In particular, Algorithm 5.1 recomputes A_λ each time a pivot changes, so it is indeed correct.

Next we prove that the list L has size at most 2^d. Suppose λ is in the range $c_i \le \lambda < c_{i+1}$. Our claim is that at most $\binom{d}{i}$ apartments are computed in this range, so that $\sum_i \binom{d}{i} = 2^d$ bounds the number of apartments in L. Write an i-sized subset σ of $[d]$ as $(\sigma_1, \sigma_2, \dots, \sigma_i)$, where $\sigma_1 < \sigma_2 \cdots < \sigma_i$. We can assign to each λ an i-sized subset σ^λ of $[d]$, where $j \in \sigma^\lambda$ if and only if the jth pivot appears in the first i columns of N_j when computing an SA-basis for M_1 and $A\Gamma_\lambda$. We can also define a well-ordering on the set of all i-sized subsets of $[d]$ lexicographically: $\sigma < \tau$ if and only if the first j with $\sigma_j \ne \tau_j$ satisfies $\sigma_j < \tau_j$. The key insight is that $\sigma^\lambda < \sigma^{\lambda+1}$ if the corresponding pivot sequences for λ and $\lambda + 1$ differ. Since there are $\binom{d}{i}$ possible choices for σ^λ, there can be at most $\binom{d}{i}$ different pivot position changes for λ in this range.

It remains to show why this key fact holds. Suppose that incrementing λ by one changes some pivot position, with the jth pivot the first to change. The above analysis shows that the jth pivot for the pair $(M_1, A\Gamma_\lambda)$ must be in the first i columns, and that this must change for the pair $(M_1, A\Gamma_{\lambda+1})$. It follows that j must be in σ^λ, and that j cannot be in $\sigma^{\lambda+1}$. Furthermore, because j

16

is the first pivot to change, for each $\ell < j$ we have $\ell \in \sigma^\lambda \iff \ell \in \sigma^{\lambda+1}$. Hence $\sigma^\lambda < \sigma^{\lambda+1}$, as desired. □

Example 5.3. Fix $K = \mathbb{Q}_3$ and the building $\mathcal{B}_5$. Let M_1 be the 5×5 identity matrix, and let each entry of M_2 and M_3 be sampled uniformly at random from the finite set $\{3^e : e \in \mathbb{Z}, -20 \le e \le 20\}$. The author took 1000 such triangles and computed enveloping membranes via Algorithm 5.1 in Mathematica. After pruning duplicate columns, the matrices describing the enveloping membranes always had at least 6 columns, and at most 25. For comparison, the upper bound implied by our Algorithm 5.1 is $5 \cdot 32 = 160$ columns. A histogram describing the frequency counts for the size of the membranes is presented in Figure 10.
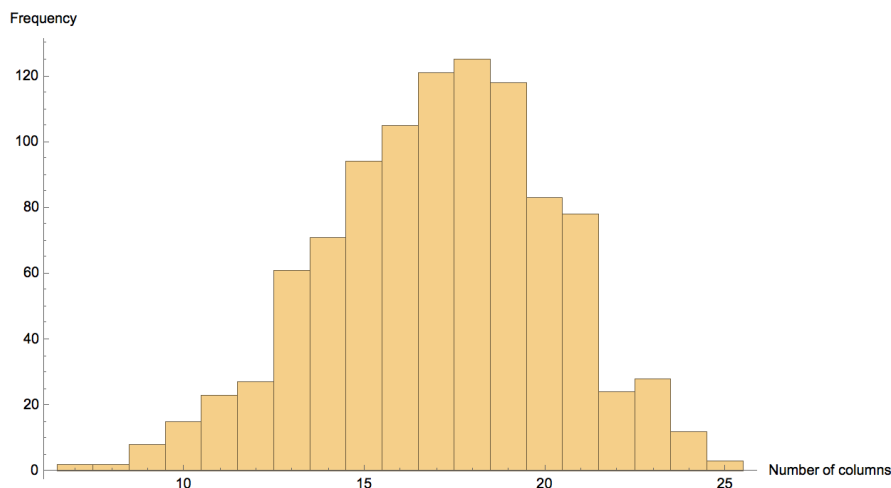

FIGURE 10. Frequency counts for the number of columns of enveloping membranes produced by Algorithm 5.1 for random convex triangles.

One example of a convex triangle attaining the maximal number of columns is given by

$$
M_2 = \begin{pmatrix} 3^{-15} & 3^{16} & 3^{-7} & 3^{-8} & 3^{-13} \\ 3^{-13} & 3^{20} & 3^{-12} & 3^{-9} & 3 \\ 3^{-19} & 3^{19} & 3^7 & 3^{-15} & 3^{10} \\ 3^9 & 3^{-12} & 3^{-12} & 3^{-17} & 3^{-18} \\ 3^{-17} & 3^{-4} & 3^{-7} & 3^{-3} & 3^{20} \end{pmatrix}, M_3 = \begin{pmatrix} 3^{-1} & 3^{-8} & 3^{-20} & 3^{-1} & 3^{-20} \\ 3^{10} & 3^6 & 3^0 & 3^2 & 3^{-20} \\ 3^{-6} & 3^8 & 3^3 & 3^5 & 3^{-13} \\ 3^{-15} & 3^9 & 3^{-9} & 3^2 & 3^{-7} \\ 3^{12} & 3^{-3} & 3^5 & 3^{-16} & 3^{-13} \end{pmatrix}.
$$

After applying Algorithm 5.1 to obtain an appropriate membrane $[M]$, the author computed the tropical polytope via Algorithm 2.19 presented in Figure 11. Note that this convex hull can be spanned by only five of the given points: $(0, 19, -8), (0, 18, 15), (0, 13, 16), (0, 12, 20)$, and $(0, 7, 20)$. Let M' be the square submatrix of M with columns corresponding to these five points. Running through Algorithm 2.19 using the apartment $[M']$ yields a coarser subdivision of the same tropical polytope. This implies that the convex hull of our three matrices M_1, M_2, and M_3 all lie in the single common apartment $[M']$, which can also be seen using [JSY07, Lemma 25]. That our algorithms do not notice this fact suggests that they likely can be improved.

REFERENCES

[AB08]   Peter Abramenko and Kenneth S. Brown, *Buildings: Theory and applications*, Graduate Texts in Mathematics, vol. 248, Springer, New York, 2008.

[CHSW11] Dustin Cartwright, Mathias Häbich, Bernd Sturmfels, and Annette Werner, *Mustafin varieties*, Selecta Math. (N.S.) **17** (2011), no. 4, 757–793.
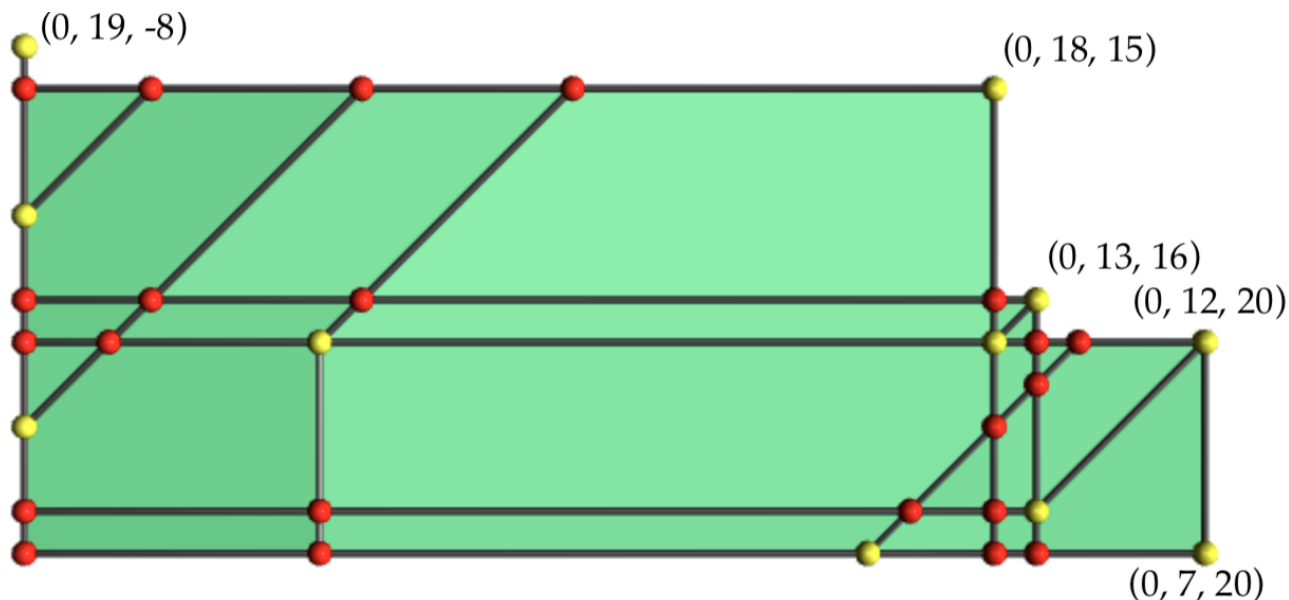
FIGURE 11. The tropical polytope isomorphic to the convex hull of M_1, M_2, M_3, with spanning vertices in yellow. Note that the x- and y-axes have been flipped.

[DS04]   Mike Develin and Bernd Sturmfels, *Tropical convexity*, Doc. Math. **9** (2004), 1–27.
[DT98]   Andreas Dress and Werner Terhalle, *The tree of life and other affine buildings*, Proceedings of the International Congress of Mathematicians, Vol. III (Berlin, 1998), vol. III, 1998, pp. 565–574.
[Fal01] Gerd Faltings, *Toroidal resolutions for some matrix singularities*, Moduli of abelian varieties (Texel Island, 1999), Progr. Math., vol. 195, Birkhäuser, Basel, 2001, pp. 157–184.
[GJ00]   Ewgenij Gawrilow and Michael Joswig, *polymake: a framework for analyzing convex polytopes*, Polytopes— combinatorics and computation (Oberwolfach, 1997), DMV Sem., vol. 29, Birkhäuser, Basel, 2000, pp. 43–73.
[Hir18] Hiroshi Hirai, *Computing degree of determinant via discrete convex optimization on euclidean building*, 2018, arXiv:1805.11245.
[Hit11] Petra Hitzelberger, *Non-discrete affine buildings and convexity*, Adv. Math. **227** (2011), no. 1, 210–244.
[HL17]   Marvin Anas Hahn and Binglin Li, *Mustafin varieties, moduli spaces and tropical geometry*, 2017, arXiv:1707.01216.
[Jos]   Michael Joswig, *Essentials of tropical combinatorics*, in preparation, `http://page.math.tu-berlin.de/~joswig/etc/index.html`, accessed in 2018.
[JSY07] Michael Joswig, Bernd Sturmfels, and Josephine Yu, *Affine buildings and tropical convexity*, Albanian J. Math. **1** (2007), no. 4, 187–211.
[KT06]   Sean Keel and Jenia Tevelev, *Geometry of Chow quotients of Grassmannians*, Duke Math. J. **134** (2006), no. 2, 259–311.
[MS15]   Diane Maclagan and Bernd Sturmfels, *Introduction to tropical geometry*, Graduate Studies in Mathematics, vol. 161, American Mathematical Society, Providence, RI, 2015.
[PRS13] Georgios Pappas, Michael Rapoport, and Brian Smithling, *Local models of Shimura varieties, I. Geometry and combinatorics*, Handbook of moduli. Vol. III, Adv. Lect. Math. (ALM), vol. 26, Int. Press, Somerville, MA, 2013, pp. 135–217.
[Rin13] Felipe Rincón, *Local tropical linear spaces*, Discrete Comput. Geom. **50** (2013), no. 3, 700–713.
[Sup08] Tharatorn Supasiti, *Serre's tree for $SL_2(\mathbb{F})$*, unpublished honors thesis, 2008.

E-mail address: `leonyz@math.berkeley.edu`