The monodromy pairing and
discrete logarithm on the Jacobian of finite graphs

Farbod Shokrieh

Communicated by Simon R. Blackburn

Abstract. Every graph has a canonical finite abelian group attached to it. This group has appeared in the literature under a variety of names including the sandpile group, critical group, Jacobian group, and Picard group. The construction of this group closely mirrors the construction of the Jacobian variety of an algebraic curve. Motivated by this analogy, it was recently suggested by Norman Biggs that the critical group of a finite graph is a good candidate for doing discrete logarithm based cryptography. In this paper, we study a bilinear pairing on this group and show how to compute it. Then we use this pairing to find the discrete logarithm efficiently, thus showing that the associated cryptographic schemes are not secure. Our approach resembles the MOV attack on elliptic curves.

Keywords. Discrete logarithm, graphs, Jacobian, monodromy pairing, generalized inverses, critical group, sandpiles.

2010 Mathematics Subject Classification. 05C50, 14H40, 11Y99.

1 Introduction

1.1 Overview

Every graph has a canonical finite abelian group whose order is the number of spanning trees of the graph. This group has appeared in the literature under many different names; in theoretical physics it was first introduced as the "abelian sandpile group" or "abelian avalanche group" in the context of self-organized critical phenomena [3, 16, 19]. In arithmetic geometry, this group appeared as the "group of components" in the study of degenerating algebraic curves [22]. In algebraic graph theory this group appeared under the name "Jacobian group" or "Picard group" in the study of flows and cuts in graphs [2]. The study of a certain chip-firing game on graphs led to the definition of this group under the name "critical group" [8, 9].

The construction of this group closely mirrors the construction of the Jacobian variety of an algebraic curve. Motivated by this analogy, Norman Biggs in [10] suggests that the Jacobian of a finite graph (which he calls the "critical group") might be suitable for discrete logarithm based cryptography.

In this paper, we study the discrete logarithm problem on the Jacobian of finite graphs. Our main result is an algorithm to efficiently compute discrete logarithms on these groups. Therefore, unlike elliptic curves and Jacobian varieties, one can not use the Jacobian of finite graphs for cryptographic purposes. It is an intriguing problem whether the fact that discrete logarithm can be done efficiently might have any algorithmic applications. Our algorithm uses a bilinear pairing, which we call the monodromy pairing, on this group. This approach is similar to the MOV attack on elliptic curves. For our application, we study the monodromy pairing and show how to compute it.

1.2 Related work

The order of the Jacobian group is the number of spanning trees of the graph [8]. Hence, the order of the group can be computed by the famous Matrix-Tree formula of Kirchhoff.

Finite graphs and algebraic curves behave similarly in many respects. Recently, there have been an increasing number of papers pursuing this analogy. Some relationship between elliptic curves and chip-firing games on graphs is noticed in [26]. In [4,5] a version of the famous Riemann-Roch theorem is proved for finite graphs, a discrete analogue of holomorphic maps between Riemann surfaces is introduced, and a graph-theoretic Riemann-Hurwitz formula is proved. A Torelli's theorem for graphs is proved in [1, 14]. The relationship between graph theory and algebraic geometry goes beyond a simple analogy. For example, Mikhalkin and Zharkov in [25] prove that an (abstract) "tropical curve" is simply a connected "metric graph".

1.3 Previous work.

Norman Biggs in [10] constructs a family of graphs with cyclic Jacobian groups, to be potentially used for cryptography. The problem of finding families of graphs with cyclic Jacobian groups is subsequently studied in [15, 23, 26]. These provide examples of cyclic Jacobian groups with appropriate order, so that discrete logarithm problem cannot be solved by the known purely group-theoretic methods.

In [11] Blackburn addresses the discrete logarithm problem for the particular family of graphs constructed by Biggs in [10]. It is fairly clear that methods presented in [11], with some minor modifications, can also be applied to the general

case. Our method is quite different from Blackburn's method, and our algorithm in §4 works for any graph.

To our knowledge, the monodromy pairing was first introduced by Bosch and Lorenzini in [13]. We have not found an easy-to-compute formula, like (3.4), in the literature.

The paper proceeds as follows. In §2 we provide the relevant definitions. The monodromy pairing is studied in §3. Using the monodromy pairing, we give our discrete logarithm algorithm in §4. Further remarks and results are outlined in §5. Appendix A contains a new proof of Theorem 3.4.

2   Definitions

2.1   Notation and terminology

Throughout this paper, a *graph* means a finite, unweighted multigraph with no loops. All graphs are assumed to be connected. For a graph G, the set of vertices is denoted by $V(G)$, and the set of edges is denoted by $E(G)$. Throughout this paper, n and m denote the number of vertices and edges, respectively.

Let $\{v_1, \ldots, v_n\}$ be an ordering of $V(G)$. With respect to this ordering, the *Laplacian matrix Q* associated to G is the $n \times n$ matrix $Q = (q_{ij})$, where q_{ii} is the degree of vertex v_i, and $-q_{ij}$ ($i \neq j$) is the number of edges connecting v_i and v_j. It is well-known (and easy to verify) that Q is symmetric, has rank $n - 1$, and the kernel of Q is spanned by $\mathbf{1}$, the all-one vector[1] (see, e.g., [7, 12]).

2.2   The Jacobian of a finite graph

Let $\mathrm{Div}(G)$ be the free abelian group generated by $V(G)$. One can think of elements of $\mathrm{Div}(G)$ as formal integer linear combination of vertices

$$\mathrm{Div}(G) = \Big\{ \sum_{v \in V(G)} a_v(v) \; : \; a_v \in \mathbb{Z} \Big\}.$$

By analogy with the algebraic curve case, elements of $\mathrm{Div}(G)$ are called *divisors* on G. For a divisor D, the coefficient a_v of (v) in D is denoted by $D(v)$.

We define by $\mathcal{M}(G) = \mathrm{Hom}(V(G), \mathbb{Z})$ the abelian group consisting of all integer-valued functions on the vertices. One can think of $\mathcal{M}(G)$ as analogous to the group $\mathcal{M}(X)^\times$ of nonzero meromorphic functions on an algebraic curve X.

For $f \in \mathcal{M}(G)$, $\mathrm{div}(f) \in \mathrm{Div}(G)$ is given by the formula

$$\mathrm{div}(f) = \sum_{v \in V(G)} \mathrm{ord}_v(f)(v),$$

---

[1] Remember that G has no loops.

where

$$\mathrm{ord}_v(f) = \sum_{\{v,w\} \in E(G)} (f(v) - f(w)).$$

Consider the group homomorphism $\deg : \mathrm{Div}(G) \to \mathbb{Z}$ defined by $\deg(D) = \sum_{v \in V(G)} D(v)$. Denote by $\mathrm{Div}^0(G)$ the kernel of this homomorphism, consisting of *divisors of degree zero*. Define $\mathrm{Prin}(G) = \{\mathrm{div}(f) \in \mathrm{Div}(G) \ : \ f \in \mathcal{M}(G)\}$ to be the group of *principal divisors*.

Lemma 2.1. $\mathrm{Prin}(G) \subseteq \mathrm{Div}^0(G)$, *and both* $\mathrm{Prin}(G)$ *and* $\mathrm{Div}^0(G)$ *are free* $\mathbb{Z}$-*modules of rank* $n - 1$.

A proof is given in [8]. As a corollary, the quotient group

$$\mathrm{Jac}(G) = \mathrm{Div}^0(G)/\mathrm{Prin}(G)$$

is well-defined and is a finite abelian group. Following [2], it is called the *Jacobian* or the *Picard*[2] group of G.

The following lemma is a direct consequence of Kirchhoff's famous *Matrix-Tree Theorem* [21] (see also [2, 8]).

Lemma 2.2. *The order of the group* $\mathrm{Jac}(G)$ *is equal to the number of* spanning trees *in G, which we denote by* $\kappa(G)$.

Following [4], for $D_1, D_2 \in \mathrm{Div}(G)$, we say that D_1 is equivalent to D_2, and write $D_1 \sim D_2$, if $D_1 - D_2$ is a principal divisor.

3 A bilinear pairing on the Jacobian of finite graphs

3.1 Generalized inverses

A matrix can have an inverse only if it is square and its columns (or rows) are linearly independent. But one can still get "partial inverse" of any matrix.

Definition 3.1. Let A be a matrix (not necessarily square). Any matrix L satisfying $ALA = A$ is called a generalized inverse of A.

It is somehow surprising that for every matrix A there exists at least one generalized inverse. In fact, more is true; every matrix has a unique *Moore–Penrose pseudoinverse*[3].

---

[2] Another appropriate notation is $\mathrm{Pic}^0(G)$.

[3] The Moore–Penrose pseudoinverse of A is a generalized inverse of A with three extra properties; see [6] for an extensive study of the subject.

Let Q be the Laplacian matrix of a connected graph. Since its rank is $n - 1$, it cannot have an inverse. But there are many ways to obtain generalized inverses:

Example 3.2. Fix an integer $1 \leq i \leq n$. Let Q_i be the $(n - 1) \times (n - 1)$ matrix obtained from Q by deleting i^{th} row and i^{th} column. Then Q_i is a full rank matrix and has an inverse Q_i^{-1}. Let $L_{(i)}$ be the $n \times n$ matrix obtained from Q_i^{-1} by inserting a zero row after the $(i - 1)^{\text{th}}$ row and inserting a zero column after the $(i - 1)^{\text{th}}$ column. Then $L_{(i)}$ is a generalized inverse of Q. One can check that

$$QL_{(i)} = I + R_{(i)},$$

where I is the identity matrix, and $R_{(i)}$ has -1 entries in the ith row and is zero everywhere else. As $R_{(i)}Q = 0$, we get $QL_{(i)}Q = Q$.

Example 3.3. Let J be the $n \times n$ all-one matrix. Then $Q + \frac{1}{n}J$ is nonsingular and $Q^+ = (Q + \frac{1}{n}J)^{-1} - \frac{1}{n}J$ is a generalized inverse of Q. In fact it is the unique *Moore–Penrose pseudoinverse* of Q; it is easy to check $QQ^+ = Q^+Q = I - \frac{1}{n}J$ and $Q^+QQ^+ = Q^+$.

These examples show that computing a generalized inverse L takes time at most $O(n^\omega)$, where ω is the exponent for matrix multiplication.

3.2 The monodromy pairing

A kind of graph-theoretic analogue of Weil pairing on the (principally polarized) Jacobian of an algebraic curve is provided by a certain bilinear pairing on $\text{Jac}(G)$, which we define in this section[4].

For D_1, D_2 in $\text{Div}^0(G)$, let m_1 and m_2 be integers such that $m_1 D_1 = \text{div}(f_1)$ and $m_2 D_2 = \text{div}(f_2)$ are principal; these exist because $\text{Jac}(G)$ is a finite group. One can easily show that

$$\frac{1}{m_2} \sum_{v \in V(G)} D_1(v) f_2(v) = \frac{1}{m_1} \sum_{v \in V(G)} f_1(v) D_2(v). \tag{3.1}$$

The pairing $\langle \cdot, \cdot \rangle : \text{Div}^0(G) \times \text{Div}^0(G) \to \mathbb{Q}$ defined by

$$\langle D_1, D_2 \rangle = \frac{1}{m_2} \sum_{v \in V(G)} D_1(v) f_2(v) \tag{3.2}$$

is symmetric and bilinear. This pairing descends to a well-defined pairing on $\text{Jac}(G)$. We use the notation $\overline{D}$ for an element of $\text{Jac}(G)$, if D is a lift of that element in $\text{Div}^0(G)$.

---

[4] The monodromy pairing is symmetric, while the Weil pairing is skew-symmetric.

Theorem 3.4. *The pairing* $\langle \cdot, \cdot \rangle : \mathrm{Jac}(G) \times \mathrm{Jac}(G) \to \mathbb{Q}/\mathbb{Z}$ *defined by*

$$\langle \overline{D}_1, \overline{D}_2 \rangle = \frac{1}{m_2} \sum_{v \in V(G)} D_1(v) f_2(v) \quad (\mathrm{mod} \ \mathbb{Z}), \tag{3.3}$$

where $m_2 D_2 = \mathrm{div}(f_2)$, *is a well-defined, symmetric, bilinear, non-degenerate pairing on* $\mathrm{Jac}(G)$.

This theorem, in a slightly different language, is proved in [13]. We give a more elementary proof in Appendix A.

Definition 3.5. We call the pairing described in Theorem 3.4 the *monodromy pairing* (see remark 1 in §5 for this terminology).

Remark 3.6. Let Φ be a finitely generated abelian group. A symmetric bilinear pairing $\langle \cdot, \cdot \rangle : \Phi \times \Phi \to \mathbb{Q}/\mathbb{Z}$ is called *non-degenerate* (or *regular*) if the group homomorphism $\Phi \to \mathrm{Hom}_{\mathbb{Z}}(\Phi, \mathbb{Q}/\mathbb{Z})$ defined by $x \mapsto \langle x, \cdot \rangle$ is injective. If it is an isomorphism, it is called *perfect* (or *unimodular*). If a pairing on a finitely generated abelian group is non-degenerate, then it is automatically perfect[5] (see [18]). For a finite abelian group Φ, this fact is immediate; there exists a (non-canonical) isomorphism between Φ and its *Pontryagin dual* $\mathrm{Hom}_{\mathbb{Z}}(\Phi, \mathbb{Q}/\mathbb{Z})$ (see, e.g., page 167 of [17]).

Let $\{v_1, \ldots, v_n\}$ be an ordering of $V(G)$. Let Q be the Laplacian matrix with respect to this ordering. This ordering gives an isomorphism between abelian groups $\mathrm{Div}(G)$, $\mathcal{M}(G)$, and the $\mathbb{Z}$-module of $n \times 1$ column vectors having integer coordinates. Under these isomorphisms the operator $\mathrm{div} : \mathcal{M}(G) \to \mathrm{Div}(G)$ coincides with the $\mathbb{Z}$-module homomorphism $Q : \mathbb{Z}^n \to \mathbb{Z}^n$. More specifically, if $[D]$ denotes the column vector corresponding to $D \in \mathrm{Div}(G)$, and $[f]$ denotes the column vector corresponding to $f \in \mathcal{M}(G)$, then $[\mathrm{div}(f)] = Q[f]$.

The given definition of the monodromy pairing in (3.3) is canonical. However, the following proposition simplifies the proof of Theorem 3.4. Moreover, it shows how one can compute the monodromy pairing in practice.

Proposition 3.7. *Let L be* any *generalized inverse of the Laplacian matrix Q. Then the monodromy pairing is given by*

$$\langle \overline{D}_1, \overline{D}_2 \rangle = [D_1]^T L [D_2] \quad (\mathrm{mod} \ \mathbb{Z}). \tag{3.4}$$

---

[5] Moreover, the group is torsion in this situation.

Proof. By definition $m_i[D_i] = [\text{div}(f_i)] = Q[f_i]$ for $i = 1, 2$. The result follows from the following computations. All equalities are mod $\mathbb{Z}$

$$
\begin{aligned}
\langle \overline{D}_1, \overline{D}_2 \rangle &= \frac{1}{m_2} \sum_{v \in V(G)} D_1(v) f_2(v) \\
&= \frac{1}{m_2} [D_1]^T [f_2] \\
&= \frac{1}{m_1 m_2} (Q[f_1])^T [f_2] \\
&= \frac{1}{m_1 m_2} [f_1]^T Q[f_2] \\
&= \frac{1}{m_1 m_2} [f_1]^T QLQ[f_2] \\
&= \frac{1}{m_1 m_2} (Q[f_1])^T L(Q[f_2]) \\
&= [D_1]^T L[D_2] \pmod{\mathbb{Z}}.
\end{aligned}
$$

$\square$

We emphasize that *any* generalized inverse of the Laplacian matrix can be used in (3.4).

4 Discrete logarithm problem on the Jacobian of a finite graph

Let $(\Phi, +)$ be a cyclic group. The *Discrete Logarithm Problem* (DLP) can be stated as:

Given $g, h \in \Phi$ with $x \cdot g = h$ for some integer x, compute $x \mod \text{ord}(g)$.

In this section we use the monodromy pairing to solve the DLP for the Jacobian of a finite graph G when $\text{Jac}(G)$ is cyclic.

In our context, we assume the elements of $\text{Jac}(G)$ are presented by some (arbitrary) lifts in $\text{Div}^0(G)$. Also, we assume[6] a generator **g** of the cyclic group $\text{Jac}(G)$ is known. We can compute and save a generalized inverse L of Q as outlined in §3.1.

---

[6] There are several efficient methods to find a generator; we omit the details here.

Algorithm 4.1 (DLP on $\mathrm{Jac}(G)$).
Input: $D, D' \in \mathrm{Div}^0(G)$ such that $\overline{D'} = x \cdot \overline{D}$ in $\mathrm{Jac}(G)$
Output: $x \mod \mathrm{ord}(\overline{D})$, the order of $\overline{D} \in \mathrm{Jac}(G)$.

(1) Compute $\langle \overline{D}, \mathbf{g} \rangle = r + \mathbb{Z}$ and $\langle \overline{D'}, \mathbf{g} \rangle = r' + \mathbb{Z}$ using formula (3.4)

(2) Solve the Diophantine equation $r' = rx + y$ (for variables $x, y \in \mathbb{Z}$) by clearing the denominators of r and r' and using the extended Euclidean algorithm, to get $x \mod \mathrm{ord}(\overline{D})$.

Analysis of the algorithm.   Since the monodromy pairing is bilinear, we have $\langle \overline{D'}, \mathbf{g} \rangle = x \langle \overline{D}, \mathbf{g} \rangle$, or $r' = rx$ in $\mathbb{Q}/\mathbb{Z}$. We still need to prove that solving the Diophantine equation precisely gives x modulo the order of $\overline{D}$ in $\mathrm{Jac}(G)$.

Lemma 4.2. *Let $\mathbf{g}$ be a generator of the cyclic group $\mathrm{Jac}(G)$. Let $\mathbf{h}$ be any element of $\mathrm{Jac}(G)$. If $\langle \mathbf{h}, \mathbf{g} \rangle = \frac{a}{b} + \mathbb{Z}$ ($a, b \in \mathbb{Z}$, $\gcd(a, b) = 1$) then b is precisely the order of $\mathbf{h}$ in $\mathrm{Jac}(G)$.*

Proof. Let γ be the order of $\mathbf{h}$ in $\mathrm{Jac}(G)$. By bilinearity of the monodromy pairing, $\frac{\gamma a}{b} + \mathbb{Z} = \gamma \langle \mathbf{h}, \mathbf{g} \rangle = \langle \gamma \cdot \mathbf{h}, \mathbf{g} \rangle = \langle \mathbf{0}, \mathbf{g} \rangle = \mathbb{Z}$, and therefore $b | \gamma$.
  On the other hand, $\langle b \cdot \mathbf{h}, \mathbf{g} \rangle = b \langle \mathbf{h}, \mathbf{g} \rangle = a + \mathbb{Z} = \mathbb{Z}$. Since $\mathrm{Jac}(G)$ is cyclic and the monodromy pairing is bilinear, all elements of $\mathrm{Jac}(G)$ must pair trivially with $b \cdot \mathbf{h}$. By non-degeneracy of the monodromy pairing we get $b \cdot \mathbf{h} = \mathbf{0}$, which means $\gamma | b$. Therefore $\gamma = b$. □

Now we can show that the algorithm precisely gives x mod the order of $\overline{D}$ in $\mathrm{Jac}(G)$; since $\overline{D'} = x\overline{D}$, order of $\overline{D'}$ divides the order of $\overline{D}$. By Lemma 4.2, for $r = \frac{a}{b}$ ($a, b \in \mathbb{Z}$, $\gcd(a, b) = 1$), b is the order $\overline{D}$ in $\mathrm{Jac}(G)$. Multiplying by b clears the denominators in $r' = rx + y$, and we get $ax + by = c$, for some integer c. It is an elementary fact that the linear Diophantine equation $ax + by = c$ (with $\gcd(a, b) = 1$) has solution, and x is determined mod b.

Running time.   The monodromy pairings in step (1) can be computed using the formula (3.4) given in Proposition 3.7. For this, we need to have a generalized inverse L; this takes time at most $O(n^\omega)$, where ω is the exponent for matrix multiplication. Notice that this computation is done only once. Each monodromy computation in step (1) can be done using $O(n^2)$ operations (multiplication and addition). Step (2) can also be done in $O(n^2)$ operations.
  For bit complexity, note that if $L = L_{(i)}$ as in Example 3.2, one can easily see that the denominators appearing in the generalized inverse are annihilated by the exponent of the Jacobian group. The exponent is bounded above by the number of

spanning trees of the graph. If we allow at most c parallel edges then there are at most $c^{n-1} \cdot n^{n-2}$ spanning trees. Therefore, the denominators can be represented by $O(n \cdot \log cn)$ bits. Moreover, one can also show that the absolute value of the entries of $L_{(i)}$ are bounded above by $R_{\max}$, the maximum effective resistance between any two vertices of the graph[7]. Therefore all integers in the algorithm can be represented in $O(n \cdot \log cn)$ bits.

Remark 4.3.

(i) The number of spanning trees of a graph on n vertices can be exponential (or more) in n. For example, Biggs in [10] constructs a family of graphs with cyclic Jacobian groups of order exponential in n. Therefore, our algorithm truly beats the known group-theoretic methods of solving DLP on finite groups.

(ii) It follows from the above discussion that, for any finite cyclic group Φ, whenever one can construct an efficiently computable perfect bilinear pairing $\langle \cdot , \cdot \rangle : \Phi \times \Phi \to \mathbb{Q}/\mathbb{Z}$, then the DLP is easy on Φ.

Remark 4.4 (DLP on the Critical group of finite graphs). Fix a vertex q. In [10] each element of the Jacobian group is presented by a canonical (relative to the base vertex q) lift in $\mathrm{Div}^0(G)$, which is called the *critical configuration* (based at q), and is defined by a certain chip-firing game on the graph. It is known that in each equivalence class of divisors there is a unique such critical configuration. *q-reduced divisors* (or *G-parking functions* based at q) provide another set of canonical elements for equivalence classes (see, e.g., [29] and references therein). Hence, the group law on the $\mathrm{Jac}(G) = \mathrm{Div}^0(G)/\mathrm{Prin}(G)$ induces a group law on the set of q-critical configurations, or the set of q-reduced divisors[8]. Biggs ([9]) calls the former set with the induced group law the *critical group* $\mathrm{K}(G)$ of the graph, and suggests in [10] that the DLP is hard on the critical group. We note that the algorithm given in this paper works for *any* lift of elements of $\mathrm{Jac}(G)$ to $\mathrm{Div}^0(G)$, and therefore it also solves the DLP on the critical group, as well as the "reduced divisors group". Some related algorithmic questions are studied in [29] and [20].

5   Concluding remarks

We conclude with some remarks.

---

[7] $R_{\max}$ is always bounded above by the diameter of the graph, but often is much smaller than the diameter.

[8] In particular, cardinality of these sets are equal to the number of spanning trees.

1. The pairing described in Theorem 3.4 is called the *monodromy* or *Grothen-dieck's* pairing for the following reason. If K is the field of fractions of a strictly Henselian discrete valuation ring R, then a theorem of Raynaud [28] asserts that the component group Φ_J of the Néron model of the Jacobian J of a semistable curve X/K is isomorphic to the Jacobian of the dual graph G of the special fiber of any semistable regular model for X over R. Under the isomorphism provided by Raynaud's theorem, the pairing on $\mathrm{Jac}(G)$ which we described in §3.2 corresponds to Grothendieck's monodromy pairing on Φ_J (see [13]).

2. By Abel's theorem for graphs (see [2]), there is a canonical isomorphism

$$\mathrm{Div}^0(G)/\mathrm{Prin}(G) \cong H_1(G,\mathbb{Z})^{\#}/H_1(G,\mathbb{Z})$$

where $H_1(G,\mathbb{Z})^{\#}$ denotes the dual of the cycle lattice $H_1(G,\mathbb{Z})$ with respect to the standard inner product on the 1-chain group $C_1(G,\mathbb{Z})$. It can be shown that under this canonical isomorphism, the monodromy pairing on $\mathrm{Div}^0(G)/\mathrm{Prin}(G)$ corresponds to the negative of the *discriminant form*[9] on $H_1(G,\mathbb{Z})^{\#}/H_1(G,\mathbb{Z})$. This and some relevant results will appear in a subsequent paper by the author.

3. Our approach to solve the DLP on the Jacobian of finite graphs resembles the MOV attack of Menezes, Okamoto, and Vanstone [24] for the DLP on elliptic curves. However, because the target group of the monodromy pairing is $\mathbb{Q}/\mathbb{Z}$ (instead[10] of $\mathbb{F}_{q^{\alpha}}$), and because of Lemma 4.2, we get a deterministic polynomial-time solution for cyclic Jacobian (instead of a probabilistic polynomial-time reduction to the DLP in the group $\mathbb{F}_{q^{\alpha}}^{\times}$).

4. If $\mathrm{Jac}(G)$ is not cyclic, then one can still use the monodromy pairing, and solve the DLP efficiently. Given a set $\{\mathbf{g}_1,\ldots,\mathbf{g}_s\}$ generating $\mathrm{Jac}(G)$, the idea is to compute $\langle \overline{D}, \mathbf{g}_i \rangle = r_i + \mathbb{Z}$ and $\langle \overline{D'}, \mathbf{g}_i \rangle = r_i' + \mathbb{Z}$ for $1 \leq i \leq s$, and then solve all Diophantine equations $r_i' = r_i x + y_i$ (for variables $x, y_i \in \mathbb{Z}$) by clearing the denominators of r_i and r_i' and using the extended Euclidean algorithm, to get x mod b_i (where $r_i = a_i/b_i$, $\gcd(a_i,b_i)=1$). Eventually, using Chinese Remainder Theorem, one can compute $x \mod (\mathrm{ord}(\overline{D}) = \mathrm{lcm}(b_1,\ldots,b_s))$. Interested reader can work out the details.

---

[9] If Λ is an integral lattice (i.e., a free $\mathbb{Z}$-module of finite rank endowed with a non-degenerate $\mathbb{Z}$-valued symmetric bilinear form), then the dual lattice $\Lambda^{\#}$ contains Λ as a finite index sub-group, and the quotient group $\Lambda^{\#}/\Lambda$ (called the *discriminant group* of the lattice) inherits in a natural way a non-degenerate $\mathbb{Q}/\mathbb{Z}$-valued symmetric bilinear form, called the discriminant form (see [27] for more details).

[10] We also note that the monodromy pairing is symmetric, while the Weil pairing is skew-symmetric.

5. We have found at least two other methods of solving the DLP in this context. One method is essentially applying the independent work of Blackburn [11] to arbitrary graphs.

6. It is worth investigating how the given solution to the DLP for the Jacobian of finite graphs can relate to the DLP for the Jacobian of algebraic curves.

7. It is an intriguing problem whether the fact that discrete logarithm can be done efficiently might have any algorithmic applications. Also the fact that the Jacobian group is actually a bilinear group with an efficiently computable pairing might have other algorithmic applications.

A   Proof of Theorem 3.4

Here we outline an elementary proof of Theorem 3.4. We choose an ordering of $V(G)$ and use the formula (3.4).

Pairing is bilinear. This is obvious!

Pairing is symmetric. This follows from (3.1). Alternatively, if L is any generalized inverse of Q then L^T is also a generalized inverse of Q (because Q is symmetric) and we have

$$\langle \overline{D}_2, \overline{D}_1 \rangle = [D_2]^T L [D_1] \pmod{\mathbb{Z}}$$
$$= ([D_1]^T L^T [D_2])^T \pmod{\mathbb{Z}}$$
$$= [D_1]^T L^T [D_2] \pmod{\mathbb{Z}}$$
$$= \langle \overline{D}_1, \overline{D}_2 \rangle.$$

Pairing is well-defined. Let D_2 and D_2' in $\text{Div}^0(G)$ be two different lifts of $\overline{D}_2 \in \text{Jac}(G)$. Then they differ by a principal divisor $[D_2'] = [D_2] + Q[g]$ for some $g \in \mathcal{M}(G)$. Let $m_1[D_1] = Q[f_1]$. Then

$$[D_1]^T L [D_2'] = [D_1]^T L [D_2] + [D_1]^T L Q[g]$$
$$= [D_1]^T L [D_2] + \frac{1}{m_1} [f_1]^T Q L Q[g]$$
$$= [D_1]^T L [D_2] + \frac{1}{m_1} [f_1]^T Q[g]$$
$$= [D_1]^T L [D_2] + [D_1]^T [g].$$

So $[D_1]^T L [D_2'] = [D_1]^T L [D_2] \pmod{\mathbb{Z}}$. By symmetry the same is true for different lifts of D_1.

Pairing is non-degenerate. We should show that if $D_1 \in \mathrm{Div}^0(G)$ be such that

$$\langle \overline{D}_1, \cdot \rangle \equiv 0 \quad (\mathrm{mod} \ \mathbb{Z})$$

then D_1 is a principal divisor. Let $\mathbf{x} = L^T[D_1] \in \mathbb{Q}^n$. If $\langle \overline{D}, \cdot \rangle \equiv 0 \ (\mathrm{mod} \ \mathbb{Z})$, then $\mathbf{x}^T \mathbf{u} \in \mathbb{Z}$, for any zero-sum column vector $\mathbf{u} \in \mathbb{Z}^n$. Substituting $\mathbf{e}_i - \mathbf{e}_1$ for $\mathbf{u}$ (where $\mathbf{e}_i$ denotes the vector with a 1 in the ith coordinate and 0's elsewhere), we get $\mathbf{x} = r\mathbf{1} + \mathbf{v}$ for some $r \in \mathbb{Q}$ and $\mathbf{v} \in \mathbb{Z}^n$ ($\mathbf{1}$ denotes the all-one vector). Multiplying by Q, we have $Q\mathbf{x} = rQ\mathbf{1} + Q\mathbf{v} = Q\mathbf{v}$ or $QL^T[D_1] = Q\mathbf{v}$. Using $m_1[D_1] = Q[f_1]$ and the fact that L^T is also a generalized inverse of Q, we get

$$QL^T[D_1] = \frac{1}{m_1} QL^T Q[f_1]$$

$$= \frac{1}{m_1} Q[f_1]$$

$$= [D_1] \,.$$

Therefore we have shown $[D_1] = Q\mathbf{v}$ for some $\mathbf{v} \in \mathbb{Z}^n$, which means D_1 must be a principal divisor.   □

Bibliography

[1] I. V. Artamkin, The discrete Torelli theorem, *Mat. Sb.* **197** (2006), 3–16.

[2] R. Bacher, P. de la Harpe and T. Nagnibeda, The lattice of integral flows and the lattice of integral cuts on a finite graph, *Bull. Soc. Math. France* **125** (1997), 167–198.

[3] P. Bak, C. Tang and K. Wiesenfeld, Self-organized criticality, *Phys. Rev. A (3)* **38** (1988), 364–374.

[4] M. Baker and S. Norine, Riemann-Roch and Abel-Jacobi theory on a finite graph, *Adv. Math.* **215** (2007), 766–788.

[5] ———, Harmonic morphisms and hyperelliptic graphs, *Int. Math. Res. Not. IMRN* (2009), 2914–2955.

[6] A. Ben-Israel and T. N. E. Greville, *Generalized Inverses*, second ed, CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC, 15, Springer-Verlag, New York, 2003, Theory and applications.

[7] N. Biggs, *Algebraic Graph Theory*, second ed, Cambridge Mathematical Library, Cambridge University Press, Cambridge, 1993.

[8] _____ , Algebraic potential theory on graphs, *Bull. London Math. Soc.* **29** (1997), 641–682.

[9] _____ , Chip-firing and the critical group of a graph, *J. Algebraic Combin.* **9** (1999), 25–45.

[10] _____ , The critical group from a cryptographic perspective, *Bull. Lond. Math. Soc.* **39** (2007), 829–836.

[11] S. R. Blackburn, Cryptanalysing the critical group: efficiently solving Biggs's discrete logarithm problem, *J. Math. Crypt.* **3** (2009), 199–203.

[12] B. Bollobás, *Modern Graph Theory*, Graduate Texts in Mathematics 184, Springer-Verlag, New York, 1998.

[13] S. Bosch and D. Lorenzini, Grothendieck's pairing on component groups of Jacobians, *Invent. Math.* **148** (2002), 353–396.

[14] L. Caporaso and F. Viviani, Torelli theorem for graphs and tropical curves, *Duke Math. J.* **153**:1 (2010), 129–171.

[15] S. Chen and S. K. Ye, Critical groups for homeomorphism classes of graphs, *Discrete Math.* **309** (2009), 255–258.

[16] D. Dhar, Self-organized critical state of sandpile automaton models, *Phys. Rev. Lett.* **64** (1990), 1613–1616.

[17] D. S. Dummit and R. M. Foote, *Abstract Algebra*, third ed, John Wiley & Sons Inc., Hoboken, NJ, 2004.

[18] A. H. Durfee, Bilinear and quadratic forms on torsion modules, *Advances in Math.* **25** (1977), 133–164.

[19] A. Gabrielov, Abelian avalanches and Tutte polynomials, *Phys. A* **195** (1993), 253–274.

[20] J. van den Heuvel, Algorithmic aspects of a chip-firing game, *Combin. Probab. Comput.* **10** (2001), 505–529.

[21] G. Kirchhoff, Über die Auflösung der Gleichungen, auf welche man bei der Untersuchung der linearen Verteilung galvanischer Ströme geführt wird, *Ann. Phys. Chem.* (1847), 497–508.

[22] D. Lorenzini, Arithmetical graphs, *Math. Ann.* **285** (1989), 481–501.

[23] _____ , Smith normal form and Laplacians, *J. Combin. Theory Ser. B* **98** (2008), 1271–1300.

[24] A. J. Menezes, T. Okamoto and S. A. Vanstone, Reducing elliptic curve logarithms to logarithms in a finite field, *IEEE Trans. Inform. Theory* **39** (1993), 1639–1646.

[25] G. Mikhalkin and I. Zharkov, *Tropical Curves, their Jacobians and Theta Functions*, Curves and abelian varieties, Contemp. Math. 465, Amer. Math. Soc., Providence, RI, 2008, pp. 203–230.

[26] G. Musiker, The critical groups of a family of graphs and elliptic curves over finite fields, *J. Algebraic Combin.* **30** (2009), 255–276.

[27] V. V. Nikulin, Integer symmetric bilinear forms and some of their geometric applications, *Izv. Akad. Nauk SSSR Ser. Mat.* **43** (1979), 111–177, 238.

[28] M. Raynaud, Spécialisation du foncteur de Picard, *Inst. Hautes Études Sci. Publ. Math.* (1970), 27–76.

[29] F. Shokrieh, *Chip-Firing Games, G-Parking Functions, and an Efficient Bijective Proof of the Matrix-Tree Theorem*, preprint (2009), http://arxiv.org/abs/0907.4761.

Author information

Farbod Shokrieh, Georgia Institute of Technology, Atlanta, Georgia 30332-0160, USA.
E-mail: shokrieh@math.gatech.edu, farbod@ece.gatech.edu