A SUBEXPONENTIAL QUANTUM ALGORITHM FOR THE SEMIDIRECT DISCRETE LOGARITHM PROBLEM
Friday 14th June, 2024

Christopher Battarbee, Delaram Kahrobaei, Ludovic Perret and Siamak F. Shahandashti
Historical Disclaimer
## Comparison with Recent Work

| Not this work\(^1,^2\) | This work |
|------------------------|-----------|
| Reduction to quantum-easy problems | Reduction to quantum-hard-ish problem |
| Works for some finite groups but not for semigroups | Works for any finite semigroup |

\(^1\)Imran and Ivanyos 2023.
\(^2\)Mendelsohn, Dable-Heath, and Ling 2023.
**Timeline**

2014-2021: design/analysis of different versions of an SDLP-based cryptosystem

**Summer 2022:** this work, first dedicated analysis of SDLP

**Spring 2023:** applications of techniques in this paper to DSS

**Christmas 2023:** faster SDLP methods in some finite groups

---

3 Habeeb, Kahrobaei, Koupparis, and Shpilrain 2014.
4 B., Kahrobaei, Perret, and Shahandashti 2023.
5 Imran and Ivanyos 2023; Mendelsohn, Dable-Heath, and Ling 2023.
SDLP
Definitions

Semidirect Product

Let $G$ be a finite semigroup and $End(G)$ its semigroup of endomorphisms. We define $G \rtimes End(G)$ to be the semigroup of pairs in $G \times End(G)$ equipped with the following multiplication:

$$(g, \phi)(h, \psi) := (g\phi(h), \phi \circ \psi)$$
**Semidirect Product**

Let $G$ be a finite semigroup and $\text{End}(G)$ its semigroup of endomorphisms. We define $G \rtimes \text{End}(G)$ to be the semigroup of pairs in $G \times \text{End}(G)$ equipped with the following multiplication:

$$(g, \phi)(h, \psi) := (g\phi(h), \phi \circ \psi)$$

Notice

$$(g, \phi)^2 = (g\phi(g), \phi^2)$$

$$(g, \phi)^3 = (g, \phi)(g\phi(g), \phi^2) = (g\phi(g)\phi^2(g), \phi^3)$$

$$(g, \phi)^4 = (g, \phi)(g\phi(g)\phi^2(g), \phi^3) = (g\phi(g)\phi^2(g)\phi^3(g), \phi^4)$$
Definitions

Semidirect Exponentiation

Fix \((g, \phi) \in G \rtimes \text{End}(G)\). Define \(s_{g,\phi} : \mathbb{N} \rightarrow G\) to be the group element such that

\[
(g, \phi)^x = (s_{g,\phi}(x), \phi^x)
\]

We have seen that

\[
s_{g,\phi}(x) = g \phi(g) \cdots \phi^{x-1}(g)
\]

SDLP

Fix \(G \rtimes \text{End}(G)\) and a pair \((g, \phi)\). Suppose we are given \(s_{g,\phi}(x)\) for some \(x \in \mathbb{N}\). The Semidirect Discrete Logarithm Problem is to recover \(x\).
Examples

Let $G = M_3(\mathbb{Z}_3)$, $A = \begin{pmatrix} 0 & 2 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \\ 0 & 0 & 2 \end{pmatrix}$, $\phi_B(M) = BMB^{-1}$.

Then
Examples

Let $G = M_3(\mathbb{Z}_3)$, $A = \begin{pmatrix} 0 & 2 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \\ 0 & 0 & 2 \end{pmatrix}$, $\phi_B(M) = BMB^{-1}$.

Then

$$s_{A,\phi_B}(2) = A\phi_B(A) = ABAB^{-1} = \begin{pmatrix} 1 & 2 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$
Examples

Let $G = M_3(\mathbb{Z}_3)$, $A = \begin{pmatrix} 0 & 2 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \\ 0 & 0 & 2 \end{pmatrix}$, $\phi_B(M) = BMB^{-1}$.

Then

$s_{A,\phi_B}(2) = A\phi_B(A) = ABAB^{-1} = \begin{pmatrix} 1 & 2 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

$s_{A,\phi_B}(3) = A\phi_B(A)\phi_B^2(A) = A(BAB^{-1})(B^2AB^{-2}) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$
Examples

Let $G = M_3(\mathbb{Z}_3)$, $A = \begin{pmatrix} 0 & 2 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \\ 0 & 0 & 2 \end{pmatrix}$, $\phi_B(M) = BMB^{-1}$.

Then

$s_{A,\phi_B}(2) = A\phi_B(A) = ABAB^{-1} = \begin{pmatrix} 1 & 2 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

$s_{A,\phi_B}(3) = A\phi_B(A)\phi_B^2(A) = A(BAB^{-1})(B^2AB^{-2}) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

$s_{A,\phi_B}(4) = A(BAB^{-1})(B^2AB^{-2})(B^3AB^{-3}) = \begin{pmatrix} 2 & 0 & 2 \\ 2 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix}$
Examples

Let $G = M_3(\mathbb{Z}_3)$, $A = \begin{pmatrix} 0 & 2 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 2 \\ 0 & 0 & 2 \end{pmatrix}$, $\phi_B(M) = BMB^{-1}$.

Then

$s_{A,\phi_B}(2) = A\phi_B(A) = ABAB^{-1} = \begin{pmatrix} 1 & 2 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

$s_{A,\phi_B}(3) = A\phi_B(A)\phi_B^2(A) = A(BAB^{-1})(B^2AB^{-2}) = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

$s_{A,\phi_B}(4) = A(BAB^{-1})(B^2AB^{-2})(B^3AB^{-3}) = \begin{pmatrix} 2 & 0 & 2 \\ 2 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix}$

$\ldots$

$s_{A,\phi_B}(10) = \begin{pmatrix} 1 & 2 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix} = s_{A,\phi_B}(2)$
A Group Action
The $\ast$ Operator

$$(s_g, \phi(x + y), \phi^{x+y}) = (g, \phi)^{x+y} = (g, \phi)^x (g, \phi)^y$$

$$= (s_g, \phi(x), \phi^x) (s_g, \phi(y), \phi^y)$$

$$= (s_g, \phi(x) \phi^x (s_g, \phi(y)), \phi^{x+y})$$

so $s_g, \phi(x + y) = s_g, \phi(x) \phi^x (s_g, \phi(y))$. We can add in the argument of $s_g, \phi$.  


The $\ast$ Operator

$$(s_g, \phi(x + y), \phi^{x+y}) = (g, \phi)^{x+y} = (g, \phi)^x(g, \phi)^y$$

$$= (s_g, \phi(x), \phi^x)(s_g, \phi(y), \phi^y)$$

$$= (s_g, \phi(x)\phi^x(s_g, \phi(y)), \phi^{x+y})$$

so $s_g, \phi(x + y) = s_g, \phi(x)\phi^x(s_g, \phi(y))$. We can add in the argument of $s_g, \phi$.

$\ast$

Let $X_{g, \phi} = \{s_g, \phi(i) : i \in \mathbb{N}\}$, and define $\ast : \mathbb{N} \times X_{g, \phi} \to X_{g, \phi}$ by

$$x \ast s_g, \phi(y) = s_g, \phi(x)\phi^x(s_g, \phi(y))$$

We have $x \ast s_g, \phi(y) = s_g, \phi(x + y)$. 
Set $\mathcal{X}_{g,\phi} = \{s_{g,\phi}(i) : i \in \mathbb{N}\}$.
Shape of $\mathcal{X}_{g,\phi}$

Set $\mathcal{X}_{g,\phi} = \{s_{g,\phi}(i) : i \in \mathbb{N}\}$.

Terminology

We call $n$ the **index**, $r$ the **period**, $\{g, \ldots, s_{g,\phi}(n - 1)\}$ the **tail**, and $\{s_{g,\phi}(n), \ldots, s_{g,\phi}(n + r - 1)\}$ the **cycle**.
## Definitions

### Finite Group Action

Let $G$ be a finite group, $X$ be a finite set and $*$ be a function $*: G \times X \to X$. The tuple $(G, X, *)$ is a **group action** if

1. $1_G * x = x$ for each $x \in X$
2. $(gh) * x = g * (h * x)$ for each $g, h \in G, x \in X$

### Vectorisation$^6$/Group Action DLog

Let $(G, X, *)$ be a group action. Given $x, y \in X$, the **vectorisation problem** is to find a $g$ (if one exists) such that $g * x = y$.

---

$^6$Couveignes 2006.
Theorem [B., Kahrobaei, Perret, Shahandashti]

Let $G$ be a finite semigroup and consider the semigroup $G \rtimes \text{End}(G)$. Fix a pair $(g, \phi) \in G \rtimes \text{End}(G)$, and let $C_{g,\phi}$ denote the corresponding cycle. The tuple $(\mathbb{Z}_r, C_{g,\phi}, \star)$ is a free, transitive group action, where $r$, the period associated to $(g, \phi)$, is $|C_{g,\phi}|$. 
Theorem [B., Kahrobaei, Perret, Shahandashti]

Let $G$ be a finite semigroup and consider the semigroup $G \rtimes \text{End}(G)$. Fix a pair $(g, \phi) \in G \rtimes \text{End}(G)$, and let $C_{g,\phi}$ denote the corresponding cycle. The tuple $(\mathbb{Z}_r, C_{g,\phi}, \star)$ is a free, transitive group action, where $r$, the period associated to $(g, \phi)$, is $|C_{g,\phi}|$.

Theorem [B., Kahrobaei, Perret, Shahandashti]

There is a fast quantum reduction from SDLP w.r.t $(g, \phi)$ to a vectorisation problem, and therefore quantum algorithms for SDLP of quantum complexity $2^\mathcal{O}(\sqrt{\log r})$, where $r$ is the period associated to $(g, \phi)$. 
The Reduction
Well-known that the Vectorisation Problem reduces to dihedral hidden subgroup problem.\textsuperscript{7}

Dihedral hidden subgroup problem admits (a) quantum algorithm with complexity $2^{O(\sqrt{\log n})}$ for $D_{2n}$.\textsuperscript{8}

Reduction of Semigroup DLog to a DLog problem has to address a similar structure to us.\textsuperscript{9}

\textsuperscript{7} Childs, Jao, and Soukharev 2014.
\textsuperscript{8} Kuperberg 2005.
\textsuperscript{9} Childs and Ivanyos 2014.
Scenario 1: $x < n$

$$s_g,\phi(1) \quad s_g,\phi(2) \quad ... \quad s_g,\phi(x) \quad ... \quad s_g,\phi(n)$$

$$s_g,\phi(n + 1)$$

$$s_g,\phi(n + r - 1)$$
Scenario 2: $x \geq n$
Roadmap Given \( n, r \)

Suppose we are given \( n, r \).
Roadmap Given $n, r$

Suppose we are given $n, r$.
Notice that $r \ast s_{g,\phi}(x) = s_{g,\phi}(x) \iff s_{g,\phi}(x) \in C_{g,\phi}$
Roadmap Given $n, r$

Suppose we are given $n, r$.
Notice that $r \ast s_{g,\phi}(x) = s_{g,\phi}(x) \iff s_{g,\phi}(x) \in C_{g,\phi}$

\[
\begin{align*}
\text{find smallest } k \text{ s.t. } & r \ast (k \ast s_{g,\phi}) \\
\text{return } n - k
\end{align*}
\]

\[
\begin{align*}
\text{obtain } k \text{ by giving } & s_{g,\phi}(n), s_{g,\phi}(x) \\
\text{to Vectorisation Problem oracle} & \\
\text{return } n + k
\end{align*}
\]
Given \( r \) compute \( n \) as the smallest integer such that
\[
r \ast s_{g,\phi}(n) = s_{g,\phi}(n).
\]
Given $r$ compute $n$ as the smallest integer such that $r \cdot s_{g,\phi}(n) = s_{g,\phi}(n)$.

$$\sum_{j=0}^{M-1} |j\rangle |s_{g,\phi}(j)\rangle \xrightarrow{\text{Observe second register}} \frac{1}{\sqrt{s_r}} \sum_{j=0}^{s_r-1} |x_0 + jr\rangle$$

Possible failure

QFT, classical post-processing
Conclusions
One can solve SDLP for \((g, \phi)\) in quantum time \(2^{O(\sqrt{\log r})}\) where \(r\) is a function of \(g, \phi\) - not much known about its size. In the generic case this remains state-of-the-art; possible that specific semigroups would yield faster results. Fast classical methods of computing \(n, r\) might give us interesting crypto.
Further Reading

Fast SDLP now resolved for all* finite groups.

https://eprint.iacr.org/2024/905

More on group-based cryptography:

http://aimpl.org/postquantgroup/

*up to constructive recognition.