ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH

MÔ HÌNH HÓA TOÁN HỌC

Bài tập lớn

Mathematical model for UTXO selection

Giáo viên: Huỳnh Tường Nguyên (htnguyen@hcmut.edu.vn)

Class: L02, Group: 5

Student members: Lê Văn Nam - 1712237

Hồ Công Sơn - 1712964

Đỗ Đăng Khôi - 1711807

Dương Văn Hiếu - 1711272

Phạm Văn Anh Dũng - 1710879

Hồ Chí Minh, 04/2019

Mục lục

1 Giới thiệu

Tiền mã hóa phi tập trung(decentralized cryptocurrency) là một tài sản kỹ thuật số(digital asset) sử dụng hệ thống mã hóa chung(cryptography systems collectively) để bảo đảm tính riêng tư và tính toàn vẹn mà không cần sự can thiệp của bên thứ ba. Tất cả các giao dịch trong hệ thống được đăng kí trên một cuốn sổ cái(ledger) gọi là blockchain, nó được cấu thành tuần tự từ các block. Và mỗi block chứa một số lượng giao dịch không đổi(unfixed number of transactions) và một mã băm(hash) của block trước, vì thế các giao dịch trong blockchain không thể bị thay đổi (immutable) và hợp lệ(valid). Một ví dụ quan trọng của loại tiền mã hóa này là Bitcoin, được giới thiệu năm 2008 và hiện tại có hơn 141 tỉ USD trên thị trường tiền mã hóa, với trung bình 229 ngàn giao dịch mỗi ngày và khoảng 183.89GB lưu trữ.

Bitcoin là một hệ thống blockchain dùng cho giao dịch, sử dụng mô hình dựa vào tài khoản(account-based model) để quản lý số dư tiền mã hóa. Chúng dùng các đầu ra của giao dịch để chi trả cho các đầu vào của giao dịch mới. Bất kì đầu ra giao dịch không là đầu vào của bất kì giao dịch nào thì được gọi là đầu ra giao dịch chưa từng chi tiêu (Unspent Transaction Output-UTXO). Số dư ví của bạn là tổng của tất cả các UTXO được gửi cho bạn. Khi một người dùng dùng tiền của người đó để giao dịch với một người khác, một giao dịch được sinh ra bởi việc chọn các UTXO của người đó làm đầu vào, và tạo các UTXO mới là đầu ra cho người nhận. Có thể thấy, chiến lược lựa chọn UTXO cho giao dịch đóng một vai trò thiết yếu trong quản lý số dư tiền mã hóa của tử ví điện tử. Một chiến lược lựa chọn UTXO tối ưu hóa phải đáp ứng các ràng buộc cứng và mục tiêu thiết yếu của ba nhóm chính là người dùng, cộng đồng và thợ mỏ(miners).

- Cộng đồng, kích thước tập UTXO lớn trở thành một vấn đề nghiêm trọng bởi vì nó làm giảm hiệu suất xử lý giao dịch và tăng cao chi phí sử dụng bộ nhớ.
- Người dùng muốn tạo ra một giao dịch có chi phí thấp và đảm bảo sự riêng tư cho các hành vi của họ.
- Thợ mỏ tập trung vào các giao dịch khai thác có phí cao nhất có thể.

2 Phát biểu vấn đề

Trong báo cáo này, chúng tôi đề xuất ra một mô hình toán học để chọn một tập UTXO trong hệ thống blockchain dùng cho giao dịch một cách có hiệu quả theo 2 mục tiêu chính.

- Mục tiêu đầu tiên là giảm thiểu kích thước giao dịch(transaction size), kết quả là, giảm thiểu phí giao dịch(transaction fee) mà người dùng(users) trả cho cho thợ mỏ(miners).
- Mục tiêu thứ hai là thu nhỏ kích thước tập UTXO(UTXO set size), do đó làm giảm không gian tìm kiếm(searching space) và chi phí tính toán(computation overhead).

Điều đáng chú ý là đề xuất của chúng tôi rõ ràng mang lại lợi ích cho người dùng và mục tiêu của cộng đồng. Thêm vào đó, chiến lược đề xuất của chúng tôi muốn các giao dịch được xác nhận nhanh nhất có thể bằng cách sử dụng mức phí phù hợp tùy thuộc vào nhu cầu của người sử dụng. Đây là một lợi ích ngầm cho các thợ mỏ.

3 Mô hình đề xuất

Mục tiêu Xác định một tập hợp con của tập UTXO có giá cả phải chăng sao cho thỏa mãn ràng buộc cứng H_1 và các ràng buộc mềm S_1.
Dữ liệu đầu vào

Các tham số đầu vào	Mô tả chi tiết
$U = \{u_1, ..., u_n\}$	Tập hợp các UTXO
$O = \{o_1, ..., o_n\}$	Tập hợp các đầu ra của giao dịch
$V^u = \{v_1^u, ..., v_n^u\}$	Tập hợp chứa giá trị của các đầu vào giao dịch
$V^o = \{v_1^o, ..., v_m^o\}$	Tập hợp chứa giá trị của các đầu ra giao dịch
$S^u = \{s_1^u, ..., s_n^u\}$	Tập hợp chứa kích của các đầu vào giao dịch
$S^o = \{s_1^o, ..., s_m^o\}$	Tập hợp chứa kích của các đầu ra giao dịch
M	Kích thước lớn nhất có thể của một giao dịch
α	mức phí của giao giao dịch($\alpha = 10^{-8} BTC$)
T	ngưỡng dust
ϵ	giá trị nhỏ nhất của thay đổi đầu ra

Bảng 1: CÁC THAM SỐ ĐẦU VÀO CỦA CÔNG VIỆC ĐÃ NÊU

Output

- Một tập hợp UTXO được chọn có thể chỉ chứa một output trùng khớp chính xác.
- Một đầu ra thay đổi(có thể có).

Ràng buộc cứng H_1

1. Một giao dịch phải có đủ giá trị để tiêu thụ.
2. Kích thước giao dịch không được vượt quá kích thước khối dữ liệu tối đa.
3. Tất cả các đầu ra giao dịch phải cao hơn ngưỡng DUST để chắc chắn rằng giao dịch này được chuyển tiếp đến mạng và được xác nhận.

Ràng buộc mềm S_1

1. Kích thước giao dịch được giảm thiểu.
2. Số lượng UTXO đã chọn được tối đa hóa để thu nhỏ kích thước nhóm UTXO.

3.1 Mô hình 1

Mô hình 1 là để giảm thiểu phí giao dịch như sau.

1. Các biến
   - Biến quyết định

$$x_i = \begin{cases} 1, & \text{nếu UTXO } u_i \text{ được chọn} \\ 0, & \text{ngược lại} \end{cases} \tag{1}$$

   - Biến trung gian:
     - y : Kích thước giao dịch.
     - z_v: Giá trị của thay đổi đầu ra.
     - z_s: Kích thước của thay đổi đầu ra.

$$z_s = \begin{cases} 0, & 0 \leq z_v \leq \varepsilon \\ \beta, & z_v > \varepsilon \end{cases} \tag{2}$$

2. Các ràng buộc
   - Kích thước giao dịch không được vượt quá kích thước khối dữ liệu tối đa.

$$y = \sum_{i|u_i \in U} s_i^u * x_i + \sum_{j|o_j \in O} s_j^o + z_s \leq M \tag{3}$$

   - Một giao dịch phải có đủ giá trị để tiêu thụ.

$$\sum_{i|u_i \in U} v_i^u * x_i = \sum_{j|o_j \in O} v_j^o + \alpha * y + z_v \tag{4}$$

   - Tất cả các đầu ra giao dịch phải cao hơn ngưỡng DUST để chắc chắn rằng giao dịch này được chuyển tiếp đến mạng và được xác nhận.

$$\forall v \in V^o, v \geq T \tag{5}$$

   - Mối quan hệ giữa giá trị đầu ra thay đổi z_v và kích thước z_s của nó được xác định như sau.

$$z_s \leq \lfloor \frac{z_v}{\varepsilon} \rfloor * \beta \tag{6}$$

     Nếu $z_v \leq \varepsilon$ thì z_s bằng 0; mặt khác thì z_s bằng β

3. Hàm mục tiêu:

$$minimize \ y \tag{7}$$

3.2 Mô hình 2

Mục tiêu của Model 2 là để tìm maximize số lượng mà UTXO được chọn để thu hẹp lại kích thước của nhóm UTXO ban đầu. Model 2 sẽ được xây dựng dựa trên kết quả thu được từ Model 1 như sau:

1. Các biến: bao gồm tất cả các biến trong Model 1
2. Các ràng buộc: bao gồm tất cả các ràng buộc trong Model 1 và thêm một ràng buộc như sau:

$$y \leq (1 + \gamma) \times Y \tag{8}$$

   - Y là min của kích thước giao dịch thu được từ Model 1
   - γ: là 1 hệ số $(0 \leq y \leq 1)$

   Nếu γ tiến đến 0, chúng ta muốn giữ lại kích thước giao dịch nhỏ nhất thu được từ kết quả của Model 1. Mặc khác, một giao dịch có kích thước phù hợp khi nó được tạo ra bởi một số lượng UTXO càng lớn càng tốt.

3. Hàm mục tiêu:

$$maximize \ (\sum_{i|u_i \in U} x_i - z_s/\beta) \tag{9}$$

4 Đánh giá mô hình

Dữ liệu đầu vào : Toàn bộ các text file(.txt) trong thư mục dataset() tải trên BKEL.
Dữ liệu đầu ra : output.txt được tạo ra bởi python script, 10 dòng đầu của output.txt :

```
// Total transaction size : 85470
// Total number of selected UTXOs : 314

//filename //status //y //z_s //z_v //epsilon //totalSelectedUTXOs
5ad448a959678302d59e6f75.txt Optimal  216 34    66581864     7359 1
5ad44bfdce94cf05c955f862.txt Optimal  364 34        1366      756 2
5ad44e0ece94cf05c955f864.txt Optimal  250 34    39454264     1369 1
5ad44e1bce94cf05c955f865.txt Optimal  364 34        1365      758 2
5ad4503ece94cf05c955f868.txt Optimal  364 34     1037718    20641 2
5ad4517cce94cf05c955f86a.txt Optimal  250 34   695247540     6187 1
```

Hiện thực : Python sử dụng PuLP package. Đoạn code chính của chương trình.

```
opt_model = LpProblem(name = "Model_1", sense = LpMinimize)

# # * Declare variables * # #
# Decision variables
X = [LpVariable(name="x_{0}".format(i), cat = LpBinary) for i in range(n)]

# Intermediate variables
sigma = LpVariable(name = "sig", cat = LpBinary)
# Size of change output
# A value of change output
z_v   = LpVariable(name = "z_v", lowBound = 0, cat = LpContinuous)

# # * Objective Function * # #
y = lpDot(S_u, X) + lpSum(S_o) + z_s

opt_model += y

# # * Constraint * # #
# A transaction size may not exceed maximum block data size
opt_model += y <= M

# A transaction must have sufficient value for consuming
opt_model += lpDot(V_u, X) == lpSum(V_o) + alpha*y + z_v

# All the transaction outputs must be higher than the dust threshold
opt_model += lpSum(V_o) >= T

# z_s = (z_v > epsilon)? beta : 0
large = sys.maxsize
opt_model += z_v + large*(1 - sigma) >= epsilon + 0.001
opt_model += z_v - large*(sigma) <= epsilon
opt_model += z_s >= sigma*beta

opt_model.solve()
```

Kết quả :

- Mô hình 1
  - Tổng kích thước giao dịch của các file dữ liệu : 85470
  - Tổng các UTXO được chọn của các file dữ liệu : 314
- Mô hình 2(gamma = 1)
  - Tổng kích thước giao dịch của các file dữ liệu : 132218
  - Tổng các UTXO được chọn của các file dữ liệu : 629

5 Kết luận

Trong bài báo này, chúng tôi đã đề xuất hai mô hình toán học cho việc giải quyết hai mục tiêu thiết yếu khi tạo giao dịch mới trên blockchain. Mô hình đầu tiên giảm thiểu kích thước giao dịch để nó có thể tạo ra một khoản phí nhỏ cho nhiệm vụ khai thác chịu trách nhiệm xác nhận giao dịch này trên mạng. Cái thứ hai được chế tạo để kiềm chế sự bùng nổ của nhóm UTXO bằng cách chọn càng nhiều càng tốt số lượng UTXO trong khi duy trì kích thước giao dịch để giúp người dùng trả chi phí phải chăng và phù hợp. Thí nghiệm của chúng tôi có cho thấy kết quả tốt hơn so với các giao dịch thực tế hiện tại, chiến lược HVF và LVF. Mặc dù mô hình đề xuất của chúng tôi là áp dụng trong thực tế, chúng ta cần thực hiện các thử nghiệm tiếp theo trên bộ dữ liệu lớn hơn để đo thời gian chạy và tài nguyên tiêu thụ cũng là hai yếu tố quan trọng, đặc biệt là để triển khai trong các thiết bị di động.

Tài liệu

[1] wikipedia. "**link: http://en.wikipedia.org/**", , truy cần lần cuối: 05/05/2019.

[2] Frey, D., Makkes, M. X., Roman, P.-L., Taiani, F., Voulgaris, S.: Bringing secure Bitcoin transactions to your smartphone. The 15th International Workshop on Adaptive and Reflective Middleware, (2016).

[3] Antonopoulos, A. M.: Mastering Bitcoin. 2nd edn. O'Reilly Media, CA 95472 (2014).

[4] Bitcoinjs: Open Source Organisation for Bitcoin JavaScript Libraries,https://github.com/bitcoinjs. Truy cần lần cuối vào 01/05/2019.

[5] Bitcoinj: Library for working with the Bitcoin protocol,https://bitcoinj.github.io. Truy cần lần cuối vào 28/04/2019.

[6] Yanovich, Y., Mischenko, P., Ostrovskiy, A.: Shared Send Untangling in Bitcoin, White paper, Bitfury Group Limited (2016).

[7] Dai, P., Mahi, N., Earls, J., Norta, A.: Smart-Contract Value-Transfer Protocols on a Distributed Mobile Application Platform, https://qtum.org/uploads/files/cf6d69348ca50dd985b60425ccf282f3.pdf, (2016).

[8] Sergi, D.-S., Cristina, P.-S., Guillermo, N.-A., Jordi, H.-J.: Analysis of the Bitcoin UTXO set, IACR Cryptology ePrint Archive, (2017).

[9] Erhardt, M.: An Evaluation of Coin Selection Strategies, Master thesis, Karlsruhe Institute of Technology, URL: http://murch.one/wp-content/uploads/2016/11/erhardt2016coinselection.pdf, (2016).

[10] Zahnentferner, J.: Chimeric ledgers: Translating and unifying utxo-based and account-based cryptocurrencies, Cryptology ePrint Archive, Report 2018/262, 2018. https://eprint.iacr.org/2018/262, (2018).

[11] Chepurnoy, A., Kharin, V., Meshkov, D.: A Systematic Approach To Cryptocurrency Fees. IACR Cryptology ePrint Archive, (2018).