An algorithmic approach to handle circular trading in commercial taxing system

Neeraj Kumar

(Roll No. CS16MTECH11010)

Under the supervision of

Dr. Sobhan Babu

A Project Report Submitted to

Indian Institute of Technology Hyderabad

In Partial Fulfillment of the Requirements for

The Degree of Master of Technology

भारतीय प्रौद्योगिकी संस्थान हैदराबाद

Indian Institute of Technology Hyderabad

Department of Computer Sience and Engineering

June, 2018

Declaration

I declare that this written submission represents my ideas in my own words, and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources that have thus not been properly cited, or from whom proper permission has not been taken when needed.

Neeraj

(Signature)

(Neeraj Kumar)

(CS16MTECH11010)

Approval Sheet

This Thesis entitled An Algorithmic Approach to handle Circular Trading in Commercial Taxation System by Neeraj Kumar is approved for the degree of Master of Technology from IIT Hyderabad.

(Dr. Ashudeb Dutta) Examiner
Dept. of EE
IITH

(Dr. Sathya Peri) Examiner
Dept. of CSE
IITH

(Dr. Sobhan Babu) Adviser
Dept. of CSE
IITH

Acknowledgements

This thesis is the end of my journey in obtaining my M.Tech. This thesis becomes the reality with kind support and could not be kept on track without the support and encouragement of several people including my well wishers, my friends, colleagues and this institution. I would like to express the deepest appreciation for all those people who made this thesis possible and an unforgettable experience for me. At the end of my thesis, it is gratifying task to express my thanks to all those who contributed in many ways for the success of this thesis and made it an extraordinary experience for me. First, I give thanks to God for protection and ability to do work. My deep gratitude goes to my supervisor **Dr. Sobhan Babu** for the guidance, encouragement and advice, he has provided throughout my research work. I am extremely lucky to have a supervisor who took keen interest about my work. Completing this work would have been more difficult without the support and friendship provided by research scholar Jithin Mathews, Priya Mehta and Suryamukhi. My family is played a crucial role to complete this thesis. This won’t be possible without love, guidance, support and encouragement of my father and mother. I am extremely thankful to them. I am also thankful for my parents and brothers for their love, prayers, caring and sacrifices for educating and preparing me for my future. I would like to thanks the staff of theory lab computer science and Aims lab for their gentle support always. My special thanks goes to my friends Rasika and Manjella for helping me in my tough times and supporting me, giving positive attitude towards academics.

Finally, I would like to convey my thanks to the Department of Computer Science and Engineering IIT Hyderabad for providing me opportunity to work at the one of greatest institution of India and meet so many interesting people.

May the Almighty God richly bless all of you.

NEERAJ KUMAR

Dedicated to

My Parents, Brothers and Sisters .

Abstract

This article presents fraudulent activity that are done by the unscrupulous desire of people to make the personal benefits by manipulating the tax in taxing system. Taxpayers manipulate the money paid to the tax authorities through avoidance and evasion activities. In this paper, we deal with a specific technique used by the tax-evaders known as a circular trading. We define an algorithm for detection and analysis of circular trade. To detect these circular trade, we have modeled whole system as a directed graph with actors being vertices and the transactions among them as directed edges. We have proposed an algorithm for detecting these circular trade. The commercial tax dataset is given by Telangana, India. This dataset contains the transaction details of participants involved in a known circular trade.

Keywords – data mining, bigdata analytics, social network analysis, circular trading, forensic accounting, value added tax

Nomenclature

VAT – Value Added Tax

₹ - Indian Rupee

Contents

Chapter 1

Introduction

In our society, fraudulent activity are immanent from the the time immemorial and these are increasing day by day. One kind of fraudulent activity in our society is very prevalent these days known as tax manipulation in taxing system. These kind of fraudulent activities are primarily motivated by the unscrupulous desire of people to make the personal benefits by exploiting the loopholes in the existing laws in a system. Some types of fraudulent activities are easier to identify. On the other hand, there are fraudulent activities that are extremely difficult to track down due to the complexity of the processes involved in handling them. Van Vlasselaer et al. gives concise and complete definition of 'fraud':

"*Fraud is an uncommon, well considered, imperceptibly concealed, time-evolving and often carefully organized crime which appears in many types of forms*."

This paper represents, a systematic technique using social network analysis to handle a complicated type of financial fraud know as a *circular trading.* This kind of fraudulent activities done by the business entities with the intention of evading tax which they are liable pay to the government. Circular trading is a theft of value added tax(VAT) from the government by creating fictitious business firms by a business entities to manipulate the financial information submitted in their commercial tax return filing.

1.1 Background and Related Work

The paper *arousal fraud* [2] by E. Dillon is another less sophisticated method used by fraudsters for tax evasion. Similarly, Bill trading [3] technique is also based on tax evasion where a dealer sells some goods to another dealer without raising an invoice, but collects the tax from him. The former dealer then issues fake invoice to a third dealer, who uses it to minimize his tax liability.

1.2 Value Added Tax

Value added tax(VAT) is a type of tax that is assessed incrementally, based on the increase in value of a product or service at each stage of production or distribution. In VAT system, when a business dealer, say dealer B, purchases some goods from another dealer, say dealer A, dealer B is liable to pay certain amount of tax to dealer A on the purchased goods to dealer A . Let us call this tax as *input tax* paid by business dealer B to A on business transaction. Similarly, when dealer B sells these goods to another dealer, say dealer C, dealer B will receive a certain amount of tax on the sold goods from dealer C and let us call it as the output tax received by dealer B from dealer C on the business transaction. The amount of tax received by the government from dealer B is the difference between the *output tax* received by dealer B and *input tax* paid by B. In other words

tax payable = (output tax received − input tax paid)

This formula is universal for any business dealer. However, when this difference gets negative the dealer receive a Credit Carry Forward(CCF) which s(he) can claim from the government or can use against paying tax in the future.

1.3 Flow of money in VAT

Manufacturer purchases raw material for ₹ 1200 from the producer who makes the raw material imposing 10% of tax and thereby collecting ₹ 120 in tax. Since, there is not any *input tax,* the tax payable to the government is ₹ 120. the manufacturer processes the raw materials, makes it into a product and sells it to a retailer for higher. Here he will collect ₹ 180 from the retailer. The amount of tax that manufacturer needs to pay to the government is *tax payable = (180 − 120) = ₹ 60.* Now, the retailer will adds more value to the product and sells to the consumer to the higher price by collecting a tax of ₹ 200. In this process, the tax payable to the government by the retailer is (200-180) = ₹ 20. So, total tax of ₹ 200 (= 120 + 60 + 20) is collected by the government from the different stages of the transaction.

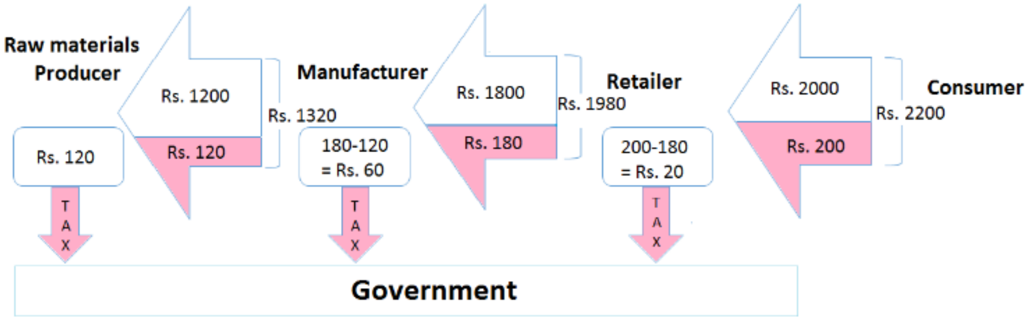


Fig. 1: Flow of money in VAT system

1.4 Circular Trading

The primary goal of circular trading is to hide the malicious sales (or) purchases information from the tax enforcement officers, and this is done by superimposing those transactions by carefully fabricated transactions. The following steps describe the classical theme of evasion:

Step1: Dealer would purposefully omit some of his/her sales and purchases information in the tax returns. These malicious tax-return information will result in the reduction of the dealer’s tax payable and he/she ends up paying less tax to the government. However, this cannot continue for longtime, since the dealer’s financial growth may not be in proportion to the amount of tax (s)he pays and consequently becomes more likely to get caught.

Step2: Guided with the intention to hide the manipulation in his/her tax returns, dealer A will create a few fictitious dealers using the personal identification details of his/her trusted acquaintances.

Step 3: At this stage, dealer A will fabricate numerous sales and purchases information between himself and the fictitious dealers by making sure that the fabricated sales and purchases information are liable to a negligible amount of tax. The tax payable on these fictitious transactions is almost zero since they amount to almost zero value addition.

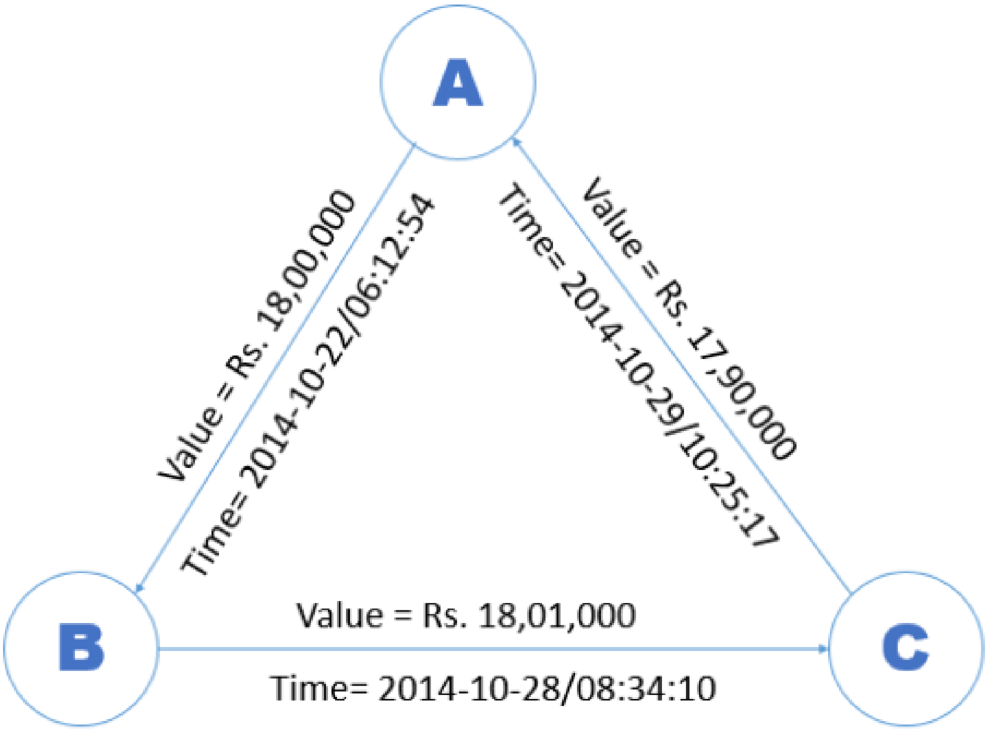

A
B
C
Value = Rs. 18,00,000
Time= 2014-10-22/06:12:54
Value = Rs. 17,90,000
Time= 2014-10-29/10:25:17
Value = Rs. 18,01,000
Time= 2014-10-28/08:34:10

Fig. 2: Circular trading

Despite of the carefully orchestrated manipulations, the dealer engaged in circular trading cannot avoid giving rise to undesired patterns in the flow of transactions. In this paper, we exploit this facet of the manipulated tax returns. One can easily observe that the manipulation, as defined in the last three steps, will result in the formation of flow of goods in a circular manner. For example, in Step 3, which is illustrated in above Figure 2, dealer A seems to sell some goods to another dealer, say to dealer B, and dealer B seems to sell the same kind of goods to dealer C, and finally dealer A purchases the same kind of goods from dealer C, hence completing the cycle. Note that the V alue of goods transferred is almost the same in all the three transactions that create the cycle. Generally, this is not a desired pattern for the flow of goods if the transactions are authentic. These cycles become much complicated to analyze with the involvement of more than 3 dealers.

The main difficulties in identifying malicious sales transactions are the large size of the dataset, complex sequences of the fictitious information and the large number of traders involved in circular trading. In this paper, we propose an algorithm to remove the fictitious transactions which are superimposed on the malicious sales transactions. This allows tax authorities to identify malicious transactions in an easy manner

The three steps detailed in this section makes the central theme for circular trading. Dealers who commit this fraud often adds up more complexity to the problem by exploiting the way VAT system works in a multi-jurisdictional trading. However, the concept of goods circling around in a cycle or a circular fashion remains the same. In [5], [6] and [7], the authors have investigated on circular trading and other related collusion techniques used in stock market trading.

Chapter 2

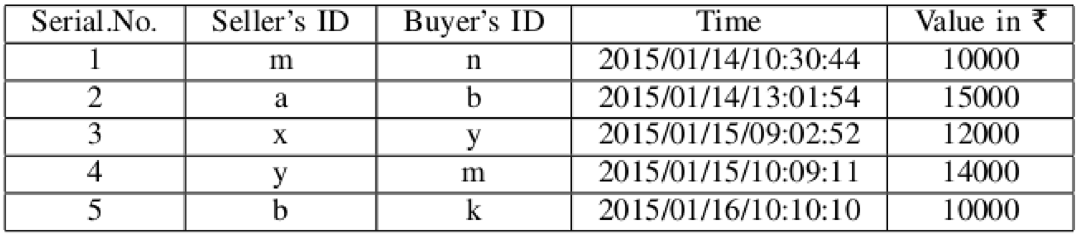
Problem Definition

In this section, we define the problem formally using graph theoretic terminologies and give a brief overview on the methodology used for handling the same. A thorough description of the algorithm along with its correctness and time complexity is discussed in the next section.

2.1 Sales transactions dataset

Table 1 shows a snapshot of the dataset used. ‘ID’ is the unique identity number of a dealer. ‘Seller’s ID’ and ‘Buyer’s ID’ shows the direction of the flow of goods, ‘Time’ gives the exact time of the transaction including the date, and the variable ‘Value’ is the amount of tax paid by the buyer to seller. For example, the second row in Table 1 can be interpreted as a dealer with ID a selling goods to a dealer with ID b on January 14th of 2015 at local time 1:01:54 pm and the buyer, dealer with ID b, gives a tax of ₹ 15, 000 to the seller.

Now we define the data-structure used in storing the above mentioned dataset. The system of all transactions among all the dealers is denoted using a weighted directed graph G = (V, E). Here V , which is the vertex set, is a set containing the ID’s of all dealers in the transactions. A transaction is defined using a weighted directed edge, and the set of all these edges are denoted by E. The weight on any edge is a 2-tuple of its corresponding ‘Value’ and ‘Time’ attribute values, (V alue, T ime). So the second row ~ with in Table 1 can be translated as a directed edge ba weight (15000, 2015/01/14/13:01:54). Note that graph G may contain multiple edges but no self loops. All multiple edges can be uniquely identified using the ‘Time’ attribute in its weight since we assume that no two transactions occur exactly at the same time.

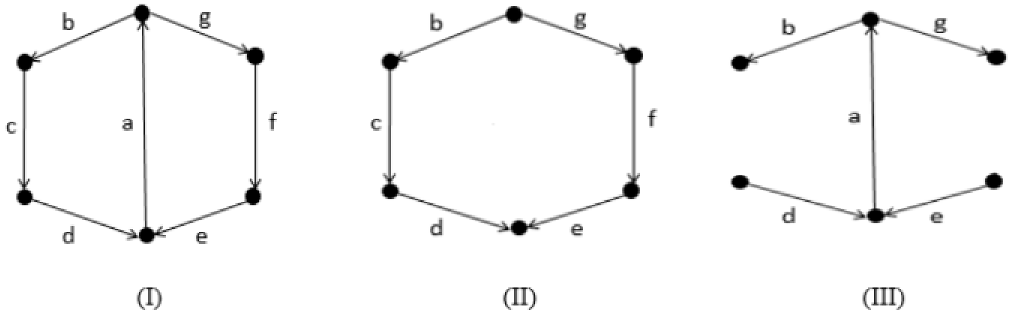


Serial.No.	Seller’s ID	Buyer’s ID	Time	Value in ₹
1	m	n	2015/01/14/10:30:44	10000
2	a	b	2015/01/14/13:01:54	15000
3	x	y	2015/01/15/09:02:52	12000
4	y	m	2015/01/15/10:09:11	14000
5	b	k	2015/01/16/10:10:10	10000

Table I: Sales transactions dataset

2.2 Cycle Deletion

As mentioned in the last section, circular trading results in the formation of undesired flow of goods in a circular fashion, which we call as cycles in graph theoretic terms. The problem of removing these cycles is important as the tax authorities can easily detect malicious transactions once the cycles are removed. Note that deleting an edge from a cycle results in the absence of that cycle from the graph. The order in which we delete cycles is significant since different order of edge deletion produces different directed acyclic graphs (DAGs) at the end. This is due to the simple fact that different cycles may share one or more edges among each other.



(I) (II) (III)

Fig 3: Cycle Deletion

For example, as illustrated in Figure 3, if a graph (given in (I)) contains two cycles that share a common edge a, viz. (a, b, c, d, a) and (a, g, f, e, a), deleting edge a results in the formation of a different DAG (as given in (II)) from the DAG formed by deleting one edge each from each cycle that is not edge e, as given in (III). Hence, we chose an ordering technique for edge deletions following the guidelines given by the taxation authorities. It is described in Observation 1.

Observation 1: In circular trading a dealer fabricates sales and purchases information between himself and the fictitious dealers such that the input tax and the output tax due to the fictitious transactions are almost the same, (i.e. tax payable on the fictitious transactions are nullified).

The *Value* parameter of the three transactions shown in Figure 2 of Section 1 illustrates Observation 1. A careful study of this observation naturally results in deleting cycles in the following particular order:

'Delete cycles in such a way that the difference between the tax values of the highest-tax-valued-edge in the cycle, (where, 'Tax value' is the second element in the 2-tuple denoting the weight of an edge), and the lowest-tax-valued-edge in the cycle is minimized.'

Using this technique, we force our algorithm to prioritize the deletion of cycles with all its edges having almost the same flow before deleting other cycles.

Chapter 3

Design and Analysis of the Algorithm

The entire technique of deleting cycles is covered in algorithms 1, 2 and 3. Algorithm 1 invokes a function defined in algorithm 2, which in turn invokes a function defined in algorithm 3. We give the complete algorithm, a brief overview of the same, along with its proof of correctness and time complexity analysis in this section. But first, let us define few terminologies:

If there exist multiple edges from vertex x to vertex y, then *max(*e_{xy}*)* denotes the edge with the maximum *Value* among all edges directed from x to y.

Critical edge of a path P or a cycle C in a graph is an edge in the corresponding path or the cycle with the minimum *Value*. We denote it by γ_P or γ_C, respectively.

Maxflow path from a vertex x to a vertex y in a graph is the path with the *Value* of its Critical edge being the maximum among all the paths from vertex x to vertex y. We denote it by μ_{xy} . Note that vertices x and y cannot be the same, in which case we have a cycle and not a path. Hence, in such cases where x = y, we consider the "path", say μ_{xx} (or μ_{yy}), to be an unreachable path with the *Value* of its *Critical edge* equals $+\infty$, i.e., Value($\gamma_{\mu xx}$) = $+\infty$.

Algorithm 1 Weighted Cycle Deletion

```
procedure WCD($\vec{G} = (V, \vec{E})$)
    ▷ $\vec{G}$ is a weighted directed graph with multiple-edges and no self-loops
    ▷ Weight on each edge is a tuple with $Value$ and $Time$, $(Value, Time)$
    ▷ Edges of graph $\vec{G}$ are stored in edge-set $\vec{E}$ in their chronological order

    Initialize $G^{'} = \emptyset$
    ▷ $G^{'} = (V^{'}, E^{'})$, hence, $(G^{'} = \emptyset) \implies (V^{'} = E^{'} = \emptyset)$

    while ($\vec{E} \neq \emptyset$) do
        $e =$ DEQUEUE($\vec{E}$)
    ▷ Edge $e$ is the least recent edge

        $E^{'} = E^{'} \cup e$
    ▷ Note that $V^{'}$ also gets updated in the process

        while ($G^{'}$ has a cycle) do        ▷ DFS is used here
            $G^{'}$ = function DELETE_CYCLE($G^{'}, e$)
        end while

    end while
    ▷ Graph $G^{'}$ now contains the desired DAG

end procedure
```

Algorithm 2 Function definition of DELETE_CYCLE()

```
function DELETE_CYCLE($G^{'}, e$)
    ▷ edge $e$ is the most recently added edge in $G^{'}$ that formed the cycle
    ▷ $Value(e)$ gives the $Value$ of edge $e$ from its ordered 2−tuple $(Value, Time)$

    Let vertex-tuple $(u, v)$ define the directed edge $e$
    ▷ $i.e.$ edge $e$ is directed from vertex $u$ to vertex $v$

    Initialize set $S = \emptyset$ and $G^{''} = G^{'}$
    ▷ $G^{''} = (V^{''}, E^{''})$, hence, $(G^{''} = G^{'}) \implies (V^{''} = V^{'}$ and $E^{''} = E^{'})$

    while ($G^{''}$ has a cycle) do    ▷ DFS is used here
        $P$ = function MAX_MIN($G^{''}, u, v$)
    ▷ $P$ denotes a path from vertex $v$ to vertex $u$

        $S = (S \cup P)$
    ▷ $S$ contains a set of ordered tuples, where each tuple denotes a path from $v$ to $u$

        Delete edge $e^{''} \in E^{''}$, where, $Value(e^{''}) \geq Value(e_{max})$
    ▷ $e_{max}$ is the edge with the largest $Value$ in $P$
    end while

    $\forall P^{'} \in S$, update $P^{'} = (P^{'} \cup \{e\})$
    ▷ Add edge $e$ to each of the ordered tuple in $S$

    Find $P_{min} \in S$ that minimizes the difference between the $Value$ of its maximum-valued-edge and minimum-valued-edge

    ▷ $e_{min}$ be the minimum-valued-edge in $P_{min}$
    Delete a flow of $Value(e_{min})$ from all the edges of $P_{min} \in G^{'}$    ▷ $i.e.$, $\forall e \in P_{min} \in G^{'}$,
                                    $Value(e) = (Value(e) - Value(e_{min}))$

    Return graph $G^{'}$

end function
```

Algorithm 3 Function definition of MAX_MIN()

```
function MAX_MIN(($G^{''}, u, v$))
    ▷ Here we use two vectors mapped to each of the vertices in $V^{''}$, $viz.$, dist[] and parent[v]
    ▷ $Value(e)$ gives the $Value$ of edge $e$ from its ordered 2−tuple $(Value, Time)$
    $\forall w \in V^{''} \setminus v$, Initialize dist[w]= $-\infty$, parent$[w] = \emptyset$,
    dist$[v] = +\infty$, parent$[v] = \emptyset$
    Insert all vertices in $V^{''}$ to Queue $Q$ in decreasing order of their dist[] values
        ▷ $\forall x \in V^{''}$ ENQUEUE$(x, Q)$ in decreasing order of dist$[x]$
    while ($Q \neq \emptyset$) do
        $ver$ = DEQUEUE$(Q)$    ▷ Delete $ver$ from $Q$, where $ver$ is the vertex with the largest dist[] value in $Q$
        Let set $N$ contains all outgoing-neighbors of $ver$
            ▷ outgoing-neighbors of a vertex $v$ are all vertices to which $v$ has an outward directed edge
        $\forall n \in N$,
        val = minimum( dist$[ver]$, $Value(e_n)$ )
            ▷ $e_n$ is the edge with the highest $Value$ among all the edges directed from vertex $ver$ to vertex $n$
        If dist$[n] <$ val then
            dist$[n]$ = val, parent$[n]$ = $e_n$
    end while
    Return the path $P$ from vertex $v$ to vertex $u$
        ▷ Path $P$ can be found by backtracking from the vertices present in parent$[u]$ to vertex $v$
end function
```

3.1 Analysis of the algorithm

Theorem 3.1: If n is the number of vertices and m is the number of edges in the input graph G given to algorithm 1, then, algorithm 1 runs in O(m + n)·m^2·log(n) time in the worst case.

Proof : In algorithm 3, if we are using a max heap for deleting the vertices with the largest dist[] value in Q, then, in the worst case it runs in O(m+n)·log(n) time. In algorithm 2, as one can easily observe, the while loop from steps 4 − 8 takes the maximum amount of time. In the worst case, it may run Step 5 for O(m) time. Hence, # algorithm 2, in the worst case runs in O(m+n)·m·log(n) time. Finally, in algorithm 1, the while loop in steps 3 − 8 may run in O(m) time in the worst case scenario were the addition of edges in Step 5 creates a cycle in almost all cases. Hence, in the worst case scenario, the total time taken by algorithm 1 is O (m+n)·m^2·log(n) .

Chapter 4

Experiments and Results

We analyzed a case in which eight dealers are doing intensive circular trading among themselves. Figure 5 shows the details of the same in the form of a directed graph with vertices denoting the dealers, and directed edges denoting the direction of transactions along with the total amount of tax paid (in lakh of ₹, 1 lakh = ₹ 1,00,000) to the seller by the buyer.

In their monthly tax return statements, all the eight dealers show huge purchases from outside the state. Legally, they should have paid heavy taxes on all these purchases. The following points illustrate a brief overview of the transactions among them.

1. The eight dealers did total purchases of ₹ 798 crores, out of which non-creditable purchases (purchases from outside the state or international imports) are ₹ 622 crores.
2. They should have paid a total tax of ₹ 31.10 crores, but they paid only ₹ 4.47 crores as VAT & interstate sales tax (also known as CST).
3. Hence, they evaded the payment of about 85% of tax.

They have done this by using the following ways:-

4. Most of the dealers have shown branch transfers (branches located in other states) which amounts to a total of ₹ 230 crores on which no tax is required to be paid.
5. They have shown questionable amount of exports totalling to ₹ 105 crores on which no tax is required to be paid.
6. They have shown questionable amount of inter state(CST) sales totalling to ₹ 111 crores on which a much lesser rate of tax (@2%) is applicable.

7. They have also shown local VAT sales of ₹ 233 crores in total on which the output tax is ₹ 11.65 crores, but have paid only ₹ 2.47 crores to the government. They could do this by raising invoices among the group members and showing Input Tax Credit (ITC). This is where circular trading comes into picture.

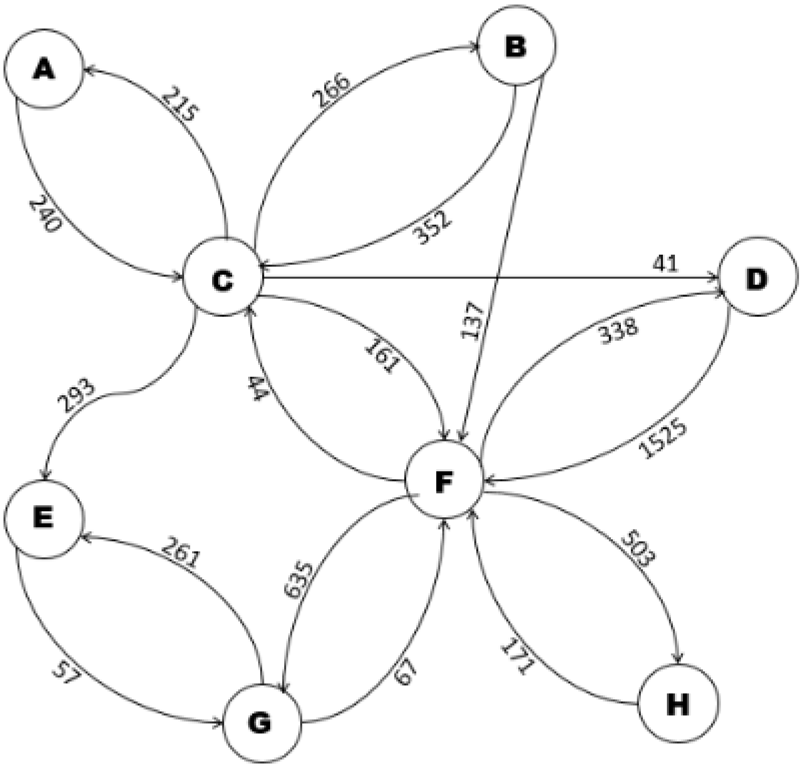


Fig 4: A know case of circular trading

Figure 5 shows the directed acyclic graph obtained after deleting all cycles from the graph given in Figure 5 using the algorithms described above. Note that the weight on each edge in the graph given in Figure 6 shows the total tax paid by a particular buyer to a particular seller (total tax is the sum of all the tax values involved in multiple transactions between them). For example, as one can observe in the edge from vertex A to vertex C in Figure 6, the total tax involved between them after deleting many transactions to remove the cycle given in Figure 5 using the proposed algorithm is

₹ 25 lakhs. It is important to note that, here the set of transactions that makes up the sum of ₹ 25 lakhs is the point of interest to tax authorities.

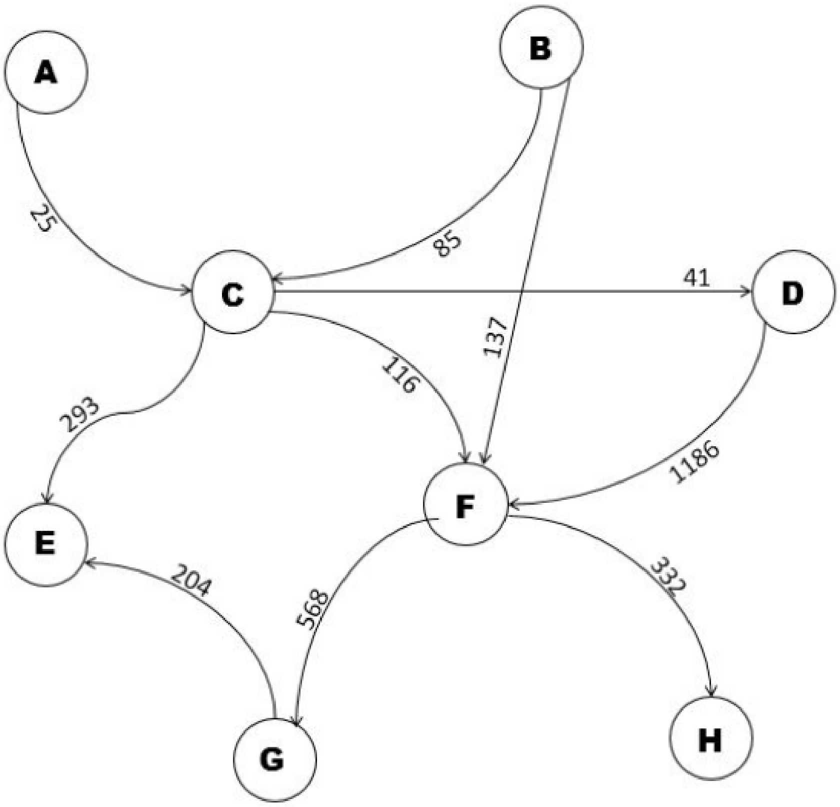


Fig 5: The Output Dag

Chapter 5

Conclusion

In this paper, we formalized the infamous tax evasion technique called circular trading. In circular trading, a group of traders do heavy fictitious sales and(or) purchase transactions in a circular manner among themselves, without any value addition, ie., the input tax and the output tax due to the fictitious transactions remains the same. The problem of removing the hence formed cycles is important as the tax authorities can easily detect malicious transactions once the cycles are removed. Here, we proposed an algorithm to remove such cycles by making use of an important observation that the amount of tax payable by a dealer due to fictitious sales and purchases transactions is almost zero.

5.1 Future Work

we try to define centrality measures for detecting the key players in circular trading. In addition, we plan to investigate whether there are more effective ways for removing cycles.

References

[1] V. Van Vlasselaer, T. Eliassi-Rad, L. Akoglu, M. Snoeck and B. Baesens. “Gotcha! Network-based fraud detection for social securityfraud,” Journal of Management Science, Jul. 2016.

[2] E. Dillon, The Fraudsters – How Con Artists Steal Your Money, Merlin Publishing, Chapter 7, Sep. 2008 A. J. Bard and L. R. Faulkner. Electrochemical Methods Fundamentals and Applications. 2nd edition. John Wiley & Sons, 2001.

[3] Godbole Committee, “Report on Economic Reforms of Jammu and Kashmir”, Ministry of Finance, Government of Jammu and Kashmir, 1998.

[4] A. Schenk and O. Oldman, Value Added Tax: A Comparative Approach, 1st ed., Cambridge Tax Law Series, Jan. 2007.

[5] G. K. Palshikar and M. M. Apte, “Collusion set detection using graph clustering,” Data Mining and Knowledge Discovery, vol. 16, Issue 2, Apr. 2008, pp. 135–164.

[6] M. Franke, B. Hoser and J. Schröder, “On the analysis of irregular stock market trading behavior,” Data Analysis, Machine Learning and Applications, Jan. 2007, pp. 355–362.

[7] K. Golmohammadi, O. R. Zaiane, and D. Daz, “Detecting stock market manipulation using supervised learning algorithms,” Data Science and Advanced Analytics, IEEE, Nov. 2014, pp. 435–441.