LARGE-SCALE MINIMUM-COST FLOW PROBLEM IN OPTIMAL MARKETING SEGMENTATION TO MAXIMIZE SHORT-TERM PROFITABILITY

Asaf SHUPO
MBNA CANADA BANK 2009

TABLE OF CONTENT

INTRODUCTION

- The purpose of this presentation is to report a minimum-cost flow (MCF) problem with a very large network structure arising in the marketing segmentation at banking system, and an efficient implementation of MCF for its solution.
- Current work is being performed in optimal marketing segmentation problem arising at MBNA Canada Bank.
- We propose an application that enables under a set of conditions required from the business area to define optimal market segmentation.

PROBLEM FORMULATION

- The banking industry regularly develops campaigns to offer current customers buying a different product or service (cross-sell).
- It is possible to build a cross-sell model to estimate the probability of an existing customer purchasing an additional product and the expected returns from that additional purchase.

PROBLEM FORMULATION CONT'

- **Which product to offer to which customer in order to maximize the marketing returns and meet the business constraints**.
- There are two ways to approach the optimal strategy:
- The short-term approach - a strategy that maximize the profits for the next period. It doesn't take into account that the customers can or will be approached again in the future.
- The long-term approach - a strategy that maximizes the total net revenues from a customer. These net revenues are often called the Customer Lifetime Value (or CLV).

5

DIFFICULTIES

- There are multiple products and it is operated under a complex set of business constraints,
- The opportunities to develop marketing campaigns are enhanced in such a way that it is now profitable to support a larger number of marketing segments which increases largely the complexity and the dimension of the segmentation and the identification of the appropriate marketing activity

6

MATHEMATICAL FORMULATION FROM SCOTIA BANK

- Let
- *xij* = 1 if customer *i* is offered product *j*, and 0 if not;
- *rij* the expected profit of offering customer *i* product *j*;
- *cij* the cost of offering customer *i* product *j*;
- *R* be the corporate hurdle rate.
- Find the *xij such* that :
- Subject to:

$$\max \sum_{ij} x_{ij} r_{ij}$$

$$\sum_{ij} x_{ij} c_{ij} \le Campaign\_budget$$

$$\sum_{i} x_{ij} \ge Minimum\_offers\_of\_product\_j$$

$$\sum_{ij} x_{ij} r_{ij} \ge \sum_{ij} x_{ij} c_{ij} (1+R) \_\_Corporate\_Hurdle$$

$$x_{ij} \in \{0,1\}$$

- This formulation captures only the bare elements of the problem
- This ideal formulation is difficult to solve because of its scale. For 1 million customers and 10 products there are 10-million integer variables *xij*, these yield $2^{10,000,000}$ possible customer-offer combinations (Storey, Marc-David Cohen, 2004).

SIMPLIFIED PROBLEM FORMULATION

- Aggregate customers based on the coefficients *rij* in the ideal formulation
- Use the clusters to aggregate customers into similar groups and use the cluster centroids as representative of the data for all the customers within a single cluster.

- Reformulate the linear program problem so that rather than assigning offers to individual customers the program identifies proportions within each cluster for each product offer.

- This can be accomplished with similar constraints to those of the ideal formulation resulting in a linear program that is much smaller and much easier to solve.

- Let *yij* be the number of customers in cluster *i* that are offered product *j*;
- Let *r'ij* be the estimated expected profit given that customer in cluster *i* is offered of product j;
- Let *c'ij* be the cost of offering a customer in cluster *i* product *j*;
- Let *R* be the corporate hurdle rate.

MATHEMATICAL FORMULATION FROM BANK OF AMERICA USA

- The same problem was solved from Bank of America in USA considering it as an integer linear programming problem the same as Scotia bank in its original formulation. In order to simplify the problem they considered a micro-segment as individuals that are eligible for the same products and have the same expected revenue for those products. In difference from Scotia Bank in the simplified formulation that find portion of cluster assigned to each product they find which micro-segment to assigned to which product.
- The following are variables used in this formulation which is the same as Scotia Bank original formulation.
- i = number of micro-segments (a micro-segment is considered as individuals that are eligible for the same products and have the same expected revenue for those products)
- j = Number of products
- N_i: Number of accounts in micro-segment "i"
- E_{ij}: is 1 whether micro-segment "i" is eligible for product "j" or 0 otherwise
- Rev_{ij}: Expected Revenue when micro-segment "i" is offered products "j"
- x_{ij} : decision variable is 1 whether to offer micro-segment "i" products "j" or 0 otherwise

MATHEMATICAL FORMULATION OF MCF

- Let *G* = (*N;A*) be a directed network consisting of a finite set of ***n*** nodes, and ***m*** directed arcs. Every arc of (*i*, *j*) ∈ *A*, is associated with a flow *xij* , a cost per unit flow *cij* , a lower and upper bound on the flow respectively *lij* and *uij* .
- Let c be a linear cost function c(x) = $\sum_{(i,j)\varepsilon A} c_{ij} x_{ij}$
- Let b be a node imbalances vector *b=(b1, b2, … bn)* such that $\sum_{i\varepsilon N} b_i = 0$
- A node *i* is called a supply node when *b*i > 0, a demand node when bi < 0, and a transshipment node when *b*i = 0.
- The minimum-cost flow problem is formulated as: finding a vector *x** *such that*

minimize $$\sum_{(i,j)\varepsilon A} c_{ij} x_{ij}$$

subject to $$\sum_{j:(i,j)\varepsilon A} x_{ij} - \sum_{j:(j,i)\varepsilon A} x_{ji} = b_i \quad \text{for all } i \in N \qquad (1.2)$$

- $0 \le l_{ij} \le x_{ij} \le u_{ij}$, for all (i, j) ∈ A (1.3)
- The constraints (1.2) are known as *flow balance* constraints.
- The problem is described in matrix notation as
- minimize $\{c^T x | Ex = b \text{ and } 0 \le l \le x \le u\}$
- where *E* is a node-arc incidence matrix having a row for each node and a column for each arc.

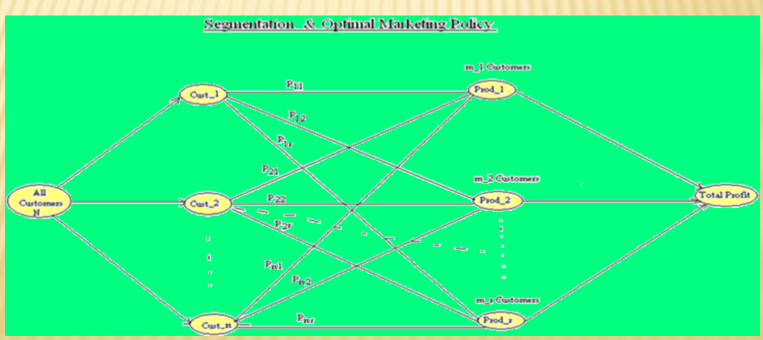
(Kelly and O'Neill, 1991)

SOLUTION METHODS

- There exist a wide variety of solution methods for optimization of network flows problem which can be categorized into *primal, dual, primal-dual, network simplex*, *scaling, relaxation*, *push-relabel, scaling push relabel* algorithms etc.
- There are a number of software that are written for different algorithms named NETFLO Primal Simplex, Cost Scaling 2 CS2, RELAX-IV Scaling Push Relabel, Network Simplex code MCF, NETFLOW2 Primal Simplex.
- CS2, RELAX-IV, and MCF are robust implementations and able to solve even large-scale minimum-cost flow problem.
- Kennington and Whitler (1998) reported on an experiment with six different codes that scaling push relabel code CS2 of Goldberg (1997) was found to be both fast and robust over a variety of problem structures.
- Our Minimum Cost Flow network optimization problem uses Cost Scaling 2 version 4.3 algorithm which performs very well in the large scale MCF structure.

IDEAL APPROACH

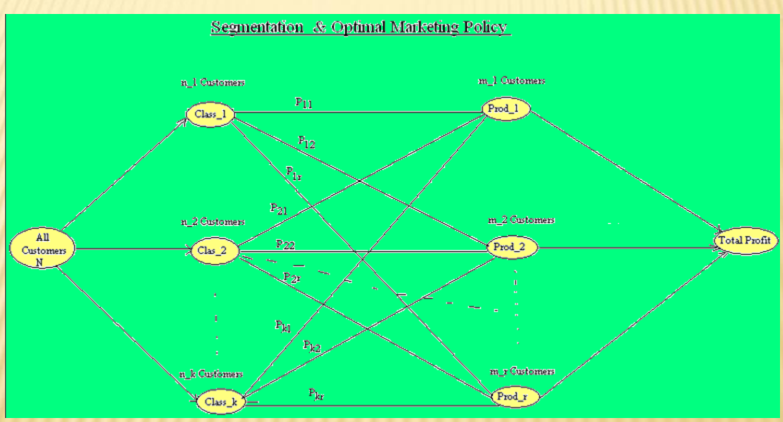
- Is the scenario in which the probability and the return that comes when a customer accepts a product are known.
- The complexity and the dimension of the MCF network problem are significantly increased. We solved this problem in the case of 2,000,000 accounts and 7 products which results in a network of 2,000,009 nodes and 15,999,976 arcs, and for 3,000,000 accounts resulting in a network of 3,000,009 nodes and 23,999,966 arcs. For 4,000,000 accounts which results in a network of 4,000,009 nodes and 31,999,957 arcs it exhausted all memory.



SIMPLIFIED APPROACH

- Clustering customers into homogeneous mini-segments (customers qualified for the same products with the same probabilities),
- Clustering customers in a mini-segment into homogeneous micro-segments (customers qualified for the same products with same profit), and
- Optimal marketing policy: which product to offer from a set of available products to each micro-segments in order to maximize the total profit.

NETWORK STRUCTURE



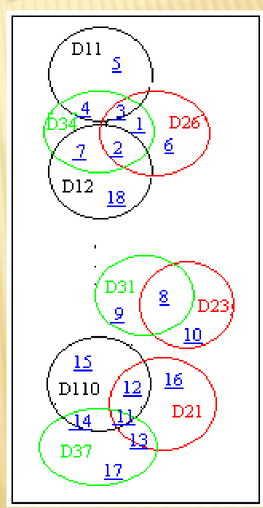
MODELS EXIST FOR EACH PRODUCT

- It is possible to create "homogeneous" mini-segments.
- A model enables creating mini-segments of customers where the probability that a customer of each mini-segment is qualified for a product is known.

CUSTOMERS SEGMENTATION IN "HOMOGENEOUS" MINI-SEGMENTS

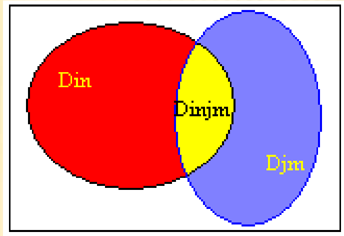
- Create mini-segments by defining every "homogeneous" intersection among all deciles for all products.
- Let denote *deciles* **i** of product **j** by ***Dij***
- *D11, D12, .., D1n1* ***n1*** deciles defined by applying the model to product ***Prod_1***
- …
- *Dr1, Dr2, .., Drnr* ***nr*** deciles defined by applying the model to ***Prod_r***
- *In an example with 3 products and 10 deciles for each product there are 18 "different" intersections among all deciles (18 mini-segments):*
- *Let pdij be the probability that customer of deciles j accept product I, then:*

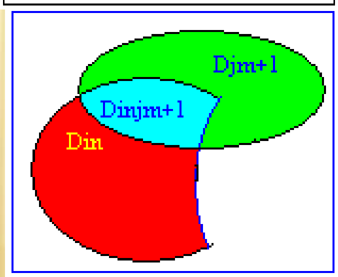
- ***Customers in mini-segment 1 accept product 2 with probability p12 = pd26 and product 3 with probability p13 = pd34,***
- ***Customers in mini-segment 2 accept product 1 with probability p21 = pd12, product 2 with probability p22 = pd26 and product 3 with probability p23 = pd34,***
- ***…***
- ***Customers in mini-segment 18 accept only product 1 with probability p181 = pd12***



CUTTING TWO DECILES

- Let be ***Din*** the decile ***n*** of product ***i*** and ***Djm*** the decile ***m*** of product ***j***.
- Cutting these two deciles results in creating three other groups with no customers in common.
- ***New Din*** the red area, new ***Djm*** the blue area and ***Dinjm*** the yellow area.
- Cutting now the new ***Din*** with ***Djm+1*** (the decile ***m+1*** of product **j)**.
- There are three groups which have no customers in common
- ***New Din*** the red area, new ***Djm+1*** the green area and ***Dinjm+1*** the blue area (cutting new ***Din*** with ***Djm+1*** is justified because all customers in ***Dinjm*** which are part of original ***Din*** are in ***Djm*** and ***Djm*** ∩***Djm+1*** = Ø).
- Any time we cut two deciles we:
- update the existing two groups reducing them by removing common customers and
- create a new combined group that contains common customers of both of them.





CUTTING PROCESS

The end of cutting process results in:

- a set groups of customers (mini-segments) who are qualified for only one product,
- a set groups of customers (mini-segments) who are qualified for only two products,

  ….
- a set groups of customers (mini-segments) who are qualified for all the products,

SEGMENTATION EXAMPLE

- In the following example it is assumed that there are 6 products and the customers are grouped into 3 groups.

- Customers in group 1 are qualified for product ***i*** with probability **p**i1,

- Customers in group 2 are qualified for product ***i*** with probability **p**i2, and

- Customers in group 3 are qualified for product ***i*** with probability **p**i3.

RESULTS OF CUTTING PROCESS

The algorithm of cutting all deciles of one product with all deciles of the other products, results in creating np (np number of products) following text files:

- ProdAccNo1.tx contains information for all accounts qualified only for 1 product
- ProdAccNo2.tx contains information for all accounts qualified exactly for 2 products
- …
- ProdAccNo6.txt contains information for all accounts qualified exactly for 6 products

- The example with 6 products and 3 groups for each product results in 728 mini-segments. For each group we have all information needed to identify a customer from which model and which group in this model he comes from, we know the probability that this customer is qualified for this product and the profit.
- In the case of 6 products and 10 deciles for each product the number of mini-segments is 864,887 which shows that this number is increased substantially with the number of groups in the model.

RESULTS

- The solution of Minimum Cost Flow problem defines the maximum flow for each arc that produces maximum total profit.
- The flow from source to 728 other nodes defines the number of customers for each mini-segment.
- The flow from segments to products defines how many customers of each segment are assigned in each product. The way how network structure is created makes it possible to generate lists for each product based on the information resulted from solution of MCF.

21

RESULTS CONT'

- With 6 products and 2,000,000 customers grouped into 10 deciles are created 864,887 mini-segments resulting in a network with 864,895 nodes and 5,534,891 arcs.
- The following are two examples of ideally scenario which was solved successfully in a PC with 2.39 GHz and 1.96 GB of RAM:
  - one with 2,000,000 customers and 7 products resulting in a network with 2,000,009 nodes and 15,999,976 arcs and.
  - another with 3,000,000 customers and 7 products the network created contains 3,000,009 nodes and 23,999,966 arcs.
- The last case with 4,000,000 customers and 7 products resulted in a network with 4,000,009 nodes and 31,999,957 arcs. This case was unable to solve in the same PC as the above two examples it exhausted all memory.

22

CONSTRAINTS

- With 6 products and 2,000,000 customers grouped into 10 deciles 864,887 mini-segments are created resulting in a network of 864,895 nodes and 5,534,891 arcs.
- The following results demonstrate that more constrains we use less profit we get they are based on synthetic data. It is supposed that the profit that comes from a customer accepts product1, product2, … product6 are respectively $10, $20, $30, $40, $50, and $60.
- The maximum profit Profit0 = $22,274,367.08 is gained in the case when there is no constrains

Product 1 flow = 46,548 customers	Expected to accept = 13,744 customers
Product 2 flow = 77,132 customers	Expected to accept = 16,975 customers
Product 3 flow = 318,088 customers	Expected to accept = 63,523 customers
Product 4 flow = 202,727 customers	Expected to accept = 29,925 customers
Product 5 flow = 864,347 customers	Expected to accept = 236,796 customers
Product 6 flow = 488,467 customers	Expected to accept = 114,351 customers

- If it is forced prod1 to be >= than 200,000 the profit1 = $21,923,292.43 < profit0 = $22,274,367.08 reduced by 1.6% of the maximum profit

Product 1 flow = 200,000 customers	Expected to accept = 50,300 customers
Product 2 flow = 49,493 customers	Expected to accept = 11,246 customers
Product 3 flow = 307,004 customers	Expected to accept = 63,281 customers
Product 4 flow = 175,534 customers	Expected to accept = 26,197 customers
Product 5 flow = 798,889 customers	Expected to accept = 229,268 customers
Product 6 flow = 466,389 customers	Expected to accept = 113,212 customers

CONSTRAINTS CONT'

- If it is forced that prod6 is ≤ than 200,000 and prod1 >= than 200,000 the result: profit2 = $19,680,819.94 < profit1= $21,923,292.43
  reduced by 11.6% of the maximum profit

Product 1 flow = 200,000 customers	Expected to accept = 50,300 customers
Product 2 flow = 54,968 customers	Expected to accept = 11,420 customers
Product 3 flow = 466,607 customers	Expected to accept = 95,173 customers
Product 4 flow = 276,845 customers	Expected to accept = 41,143 customers
Product 5 flow = 798,889 customers	Expected to accept = 229,268 customers
Product 6 flow = 200,000 customers	Expected to accept = 49,896 customers

- If it is required all high capacity be equal profit3 = $16,514,364.36 < profit2 = $19,680,819.94
  reduced by 25.9% of the maximum profit

Product 1 flow = 332,884 customers	Expected to accept = 79,799 customers
Product 2 flow = 332,884 customers	Expected to accept = 73,963 customers
Product 3 flow = 332,884 customers	Expected to accept = 68,800 customers
Product 4 flow = 332,884 customers	Expected to accept = 47,328 customers
Product 5 flow = 332,884 customers	Expected to accept = 105,713 customers
Product 6 flow = 332,889 customers	Expected to accept = 83,410 customers

CONCLUSION

- We presented an application to a real-world optimal marketing policy problem based on the Minimum Cost Flow problem containing many network flow instances, of extremely large-scale.
- Advantages of the minimum cost flow problem:
  - Integer solutions if c, b, l and u are integers, an optimal solution to its MCF will be integer.
  - Very fast solution methods
  - Is very expressive in modeling
- It is flexible and easy to use under a set of conditions required from business area
- It is able to determine not only which customers to assign in each product in order to optimize all the profit but also finds for each product how many customers associated with this product would be expected to accept this product.
- Its solution expected to produce a significant better profit than competing solution
- It can implement several business constraints simultaneous which is impossible by ad hoc techniques

FUTURE WORK

- The future work will consist in incorporating linear capacity constraints on some of the arcs, especially in the last arcs which link products with the sink. Sometime the business is interested not only on the maximum profit, not even only in the number of customers associated with each product but also with the “quality” of those customers.

- For example in the exercise where for each product is required the same number of customers, the flow for product 4 and 5 is equal 332,884 customers but for product 4 the number of customers that are expected to accept this product is 47,328 customers less than the same number for the product 5 which is 105,713 customers. This shows that the “quality” of customers associated with product 5 is better than the “quality” of the customers of product 4.

- It will be useful for business area to set constraints on the expected customers for a certain product. For example finding a solution in which the expected customer for product 4 is greater than a fixed number for example 70,000 customers.

- For our application this kind of constraints are not possible to be included in the problem they are linear constraints on these arcs.

REFERENCES

- Andrew V. Goldberg (1997). An Efficient Implementation of a Scaling Minimum-Cost Flow Algorithm. JOURNAL OF ALGORITHMS 22, 1-29 1997. ARTICLE NO. AL950805
- Andrew Storey, Marc-David Cohen (2004). Exploiting Response Models – Optimizing Cross-Sell and Up-Sell Opportunities in Banking. Information Systems Vol 29, Issue 4. ISSN:03064379
- Damian J. Kelly, B.Comm and Garrett M. O'Neill, B.Comm (1991). The Minimum Cost Flow Problem and The Network Simplex Solution Method. A dissertation presented to the National University of Ireland
- J. Kennington and J. Whitler (1998). Simplex versus cost scaling algorithms for pure networks: An empirical analysis. Technical Report 96-CSE-8, Department of Computer Science and Engineering, Southern Methodist University, Dallas, TX.
- Jeffery L. Kennigton and Richard V. Helgason (2008). Minimum Cost Network Flow Algorithms Springer US, ISBN 978-0-387-30662-9 , 2008