VOL. 10, NO. 15, AUGUST 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

AN OPTIMIZATION APPROACH FOR ENERGY-AWARE TRAFFIC ENGINEERING

Débora A. O. Santos, Luiz A. R. da Silva Mello and Marco A. Grivet Mattoso Maia
Pontifícia Universidade Católica do Rio de Janeiro – PUC-Rio Rio de Janeiro, Brazil
E-Mail: deborasantos@petrobras.com.br

ABSTRACT

The growing and incessant energy consumption has already become a global concern, and currently more than forty countries are involved in research and programs in order to create mechanisms to save it. This work deals with the Energy-aware Traffic Engineering problem applied to the backbone of an IP network, presenting an approach to select routers and circuits to be switched off in time periods where network traffic is low. The approach is to focus on the optimization formulation, rather than on heuristic methods as usually observed in the literature. The associated linear optimization program, although frequently very large, can be broken by decomposition techniques into several low size optimization problems thus bringing the problem to the realm of practical feasibility.

Keywords: traffic engineering, decomposition, benders, green networks.

INTRODUCTION

The phenomenon of Internet has turned the world into a global village, where people are more and more connected. This success was due in large part to the explosion in transmission control protocol over Internet protocol (TCP/IP) applications, which make it possible sharing information, and enjoying multimedia applications (such as high-performance multi-player games, streaming of HD movies, video conferences, Internet telephony and digital audio) and systems that require a lot of bandwidth (Iyer *et al.* 2003), decreasing distances and bringing people together.

In order to support this convergence scenario, IP backbone networks are engineered for high availability and resilience to multiple link and router failures.

So, it is easy to realize that, in despite of the positive aspects, the exponential growth use of the Internet and network resources to support it has been accompanied by negative consequences, such as a considerable increase in its energy consumption.

According to (Amaldi *et al.* 2013), in 2007, the total energy demanded by the Internet was about 900 billions kWh, representing, at that time, 5.5% of the world's electricity, with an estimated annual growth rate around 20 and 25%. Furthermore, the energy efficiency of the Internet (in other words, the ratio between the consumed energy and the total amount of data traffic) is very low, about 8 to 10 times lower than that of wireless networks (Lambert *et al.* 2012).

Several segments have participation in this consumption, but the main one belongs to the telecommunications operators.

According to recent estimates, the worldwide electricity consumption of telecom operators was, in 2012, 260TWh/y, accounting for almost 3% of the total worldwide consumption. And this number only tends to increase.

This has been a concern of the recent past, which has been intensified in recent years mainly due to energy crises in many countries all around the world.

We cannot forget that, besides the issue of energy waste, energy overconsumption leads to another very important problem that is the environmental pollution increase, due to increasingly carbon dioxide emissions in daily telecommunications network operations.

For these reasons, the concerns about energy savings in IP networks cannot be neglected anymore, and have become of great interest in the scientific community, motivating lots of studies related to Green Networks.

In this paper we focus our attention on energy management of the network and its elements, exploiting the knowledge that traffic routing is of paramount relevance in the management of any communication network.

All routing protocol is based on the simple idea that packets are forwarded on IP links along the shortest route between its source and destination nodes, thus giving rise to a class of protocols named as SPF (Shortest Path First) whose a well-known example is OSPF (Open Shortest Path First). Needless to say that the network administrator can manage this packet's routing by supplying the so-called "administrative weights" of IP links, which specify the link lengths that are used by the routing protocol for their shortest path computations. This is the only resource the network administrator can use to influence and control the routing process. Thus one of the main tasks when planning such a networks is to find administrative link weights that induce a globally efficient traffic routing configuration (Koster, *et al*. 2009), (Amaldi *et al.* 2013).

The selection of the adequate set of network parameters that ensure an optimal behavior as far as energy consumption is concerned can be commonly modeled by the formulation of a mixed-integer linear optimization problem.

The mathematical formulation of this optimization problem is simple and can be found in many articles and books about telecommunications networks (Koster, *et al*. 2009), (Amaldi *et al.* 2013).

Sadly, even for networks of moderate size, this formulation involves a extremely high number of decision variables and imposed constraints, thus requiring, in some cases, computer power and time far beyond any practical values. In the formulation presented in the articles and books above cited, it can be shown that for a network with N nodes and L links, the number D of the decision variables and the number C of the constraints is given by.

$$D = 2 \cdot N \cdot (N + L) + 2 \cdot L + N + 1 \quad (1)$$

$$C = N^2 + L + 5 \cdot L \cdot (N - 1) \quad (2)$$

For instance, for a network with 60 nodes and 80 links, the number of decision variables is 17021, and the number of constraints is 27280, figures far above reasonable levels.

This paper formulates and presents a method capable of minimizing energy consumption in a network in feasible time. The sparse nature coupled with block-structure of the constraint matrices do suggest that decomposition techniques are an effective means to reduce computer complexity and processing time. Furthermore, the decision variables do include those required for network operation such as the “administrative weights” of IP links previously mentioned.

In fact, the proposed solution involves a sequence of three stages where the Benders’ algorithm, classic in the decomposition of problems involving complicating variables, is fully deployed. At the end, only linear optimization problems with size close to the number of network links are involved and their solutions are not a computational burden even for solvers of moderate complexity.

ENERGY-AWARE TRAFFIC-ENGINEERING

Network model and problem formulization

Let’s consider the network topology represented by a directed graph $G = (V, E)$, where V corresponds to the set of nodes (or routers) and E corresponds to the set of links (or circuits). The cardinalities of V and E are respectively represented by nV and nE. We assume that the network’s Interior Gateway Protocol (IGP) is OSPF, where traffic balance is governed by Equal Cost Multi-Path (ECMP) routing, where traffic is evenly dispatched through all shortest paths to the same destination node (Koster et al., 2009).

For the sake of clearness, some variables and constants are needed to be defined and they are presented on Tables-I and II.

In addition let P_e^E be the power consumed by link e and P_t^V be the power consumed by node t. The objective function of interest here is the total energy consumption. The complete formulation of the linear program meant to be solved is shown below:

$$\min \quad z = \sum_{e \in E} P_e^E \cdot \alpha_e + \sum_{t \in V} P_t^V \cdot \beta_t \quad (3)$$

subject to:

$$\sum_{e \in \delta^-(t)} x_{et} = D_t \qquad t \in V \quad (4)$$

$$\sum_{e \in \delta^+(t)} x_{et} - \sum_{e \in \delta^-(t)} x_{et} = d_{vt} \quad (5)$$
$$(v,t) \in V,\ t \neq v$$

$$\sum_{t \in V \backslash a(e)} x_{et} \le Z \cdot C_e \qquad e \in E \quad (6)$$

$$0 \le x_{et} \le D_t \cdot u_{et}$$
$$e \in E,\ t \in V,\ t \neq a(e) \quad (7)$$

$$z_{a(e),t} - x_{et} \ge 0 \qquad e \in E,\ t \in V,\ t \neq a(e) \quad (8)$$

$$z_{a(e),t} - x_{et} \le D_t \cdot (1 - u_{et}) \quad e \in E,\ t \in V,\ t \neq a(e) \quad (9)$$

$$w_e + r_{b(e),t} - r_{a(e),t} \ge 1 - u_{et} \quad e \in E,\ t \in V,\ t \neq a(e) \quad (10)$$

$$w_e + r_{b(e),t} - r_{a(e),t} \le M \cdot (1 - u_{et})$$
$$e \in E,\ t \in V,\ t \neq a(e) \quad (11)$$

$$\alpha_e \le \beta_{a(e)} \qquad e \in E \quad (12)$$

$$\alpha_e \le \beta_{b(e)} \qquad e \in E \quad (13)$$

$$u_{et} \le \alpha_e \qquad e \in E \quad (14)$$

$$r_{st} \ge 1 \qquad (s,t) \in V \quad (15)$$

Equations (4) and (5) describe the classical Flow Preservation constraints while inequality (6) expresses the avoidance of link capacities violation. Restrictions (7), (8) and (9) are those that preserve the ECMP rules, while restrictions represented by (10) and (11) guarantee that administrative link weights w, when used by the OSPF protocol, generate shortest paths between nodes in accordance with this problem solution. Equations (12) and (13) do describe the relation between node and link on-off states while (15) states the minimum value for the path lengths.

Table-1. Summary of variables.

Notation	Description
x_{et}	ECMP flow through link e destined to node t
z_{st}	Common value of the ECMP flow destined to node t
r_{st}	Length of the w-shortest path from node s to node t
w_e	Weight assigned to link e
Z	Maximum over link utilization factors
u_{et}	Binary variable indicating if link e is on a shortest path to node t
α_e	Binary variable that indicates whether circuit e is active or not
β_t	Binary variable that indicates whether node t is active or not

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-2. Summary of parameters.

Notation	Description
V	Set of backbone routers
E	Set of backbone links
nV	Cardinality of set V
nE	Cardinality of set E
$\delta^-(v)$	Set of incoming links into node v
$\delta^+(v)$	Set of outgoing links from node v
$\#\delta^+(v)$	Cardinality of $\delta^+(v)$
$a(e)$	Source node for link e
$b(e)$	Destination node for link e
D_t	Total traffic to node t
d_{vt}	Total traffic from node v to node t
C_e	Link e capacity
M	A large enough constant, not less than the difference in length of any two paths in the network graph.

Due to the binary nature of the variables u, α_e and β_t, this problem is classified as a Mixed-Integer Linear Program (MILP). Although expressions (3) – (15) are easy to understand, the optimization problem of interest can be transformed into another one, formally much simpler, as below presented:

$$\min \quad z = \underline{c}^T \cdot \underline{y} \tag{15}$$

subject to:

$$A \cdot \underline{y} = \underline{r} \tag{16}$$

$$B \cdot \underline{y} \leq \underline{s} \tag{17}$$

$$\underline{y}_L \leq \cdot \underline{y} \leq \underline{y}_H \tag{18}$$

where decision variable $\underline{y}$ is a suitable aggregation of variables $\underline{x}, \underline{z}, \underline{r}, \underline{w}, \underline{u}, \underline{\alpha}$ and $\underline{\beta}$. A and B are matrices that describes the constraints and a glimpse of them can be observed in Figure-1 where all constraints indicated in (4) – (15) are visually represented.
From this figure is self-evident that this particular nature of A and B matrices are not only very sparse but fully made of block matrices, thus suggesting that decomposition techniques can be deployed in the hope that the original problem can be break into several others with lower dimension.

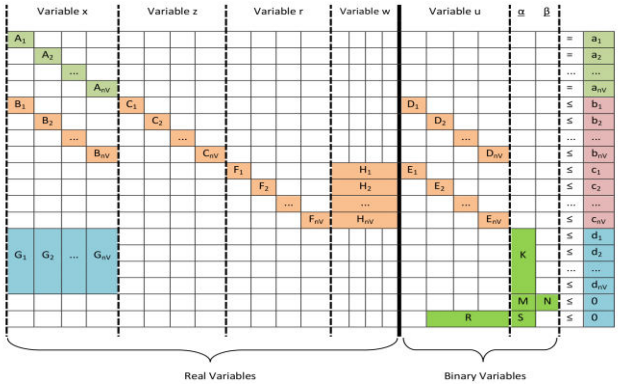

Figure-1. Visualization of the MILP constraints.

Decomposition techniques

The decomposition method proposed by Benders in 1962 is an algorithm that allows you to solve certain optimization problems that presents a block-diagonal structure in its constraints whose separable solution is prevented by those called “complicating variables”.

The central idea of Benders decomposition method, whose algorithm is described in (Benders, 1962) and presented in the Appendix, is to split the original problem into two subproblems that are interactively solved in such a way that the solution of one is inserted into the other, thus creating a sequence of alternate optimization solutions that, in the end, converge to the solution of the original problem. In a nutshell, size is dealt with by a sequence of several smaller problems.

The two simpler subproblems in which the original problem is split are known as:

(i) the **master problem**, which consists in a relaxed version of the original problem, having the set of integer variables together with their respective restrictions and a further continuous variable; and
(ii) the **slave problem**, which is the original version of the problem with the values of the integer variables temporarily fixed by the master problem.

Consider the situation where the optimization problem described by (16) – (19) can be conveniently rewritten as:

$$\min_{\underline{y1},\underline{y2}} \quad z = \underline{c}_1^T \cdot \underline{y}_1 + \underline{c}_2^T \cdot \underline{y}_2$$

subject to:

$$A_1 \cdot \underline{y}_1 + A_2 \cdot \underline{y}_2 = \underline{r}$$

$$B_1 \cdot \underline{y}_1 + B_2 \cdot \underline{y}_2 \leq \underline{r}$$

$$\underline{y}_{1,L} \leq \cdot \underline{y}_1 \leq \underline{y}_{1,H}$$

$$\underline{y}_{2,L} \leq \cdot \underline{y}_2 \leq \underline{y}_{2,H}$$

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Generally speaking, Benders' algorithm can be defined in four steps that are below introduced and discussed:

- **STEP 0: Initialization** ⇒ Variable $\underline{y}_2$ together with an artificial variable σ are conveniently initialized;
- **STEP 1: Subproblem Solution** ⇒ Variable $\underline{y}_1$ is evaluated by means of an optimization problem where the variable $\underline{y}_2$ is assumed to be known;
- **STEP 2: Convergence Checking** ⇒ Here a simple calculation is made to check if the solution so far obtained is close to the optimum one; and
- **STEP 3: Master Problem** ⇒ An optimization problem is defined to obtain new values for variables $\underline{y}_2$ and σ, considering that variable $\underline{y}_1$ is assumed to be known.

It can be observed in Figure-1 that the most convenient way of splitting the decision variables is to define the binary variables as the complicating variable $\underline{y}_2$ and the remaining variables as $\underline{y}_1$.

Hence Benders' **STEP 3** is related to an all-binary optimization problem while Benders' **STEP 1** is an all-real optimization problem. For the sake of clearness, this algorithm will be denoted as BENDERS-0.

The sizes of the variables $\underline{y}_1$ and $\underline{y}_2$ do reveal that **STEP 1** is the critical one due to its size. Nevertheless, a closer look on Figure 1 reveals that the problem at this step can be solved by a new Benders' algorithm (here called BENDERS-1) where a new "complicating variable" can be separated from the rest. It seems pretty obvious that variable $\underline{w}$ is the clear candidate to this new complicating variable. The corresponding **STEP 1** of this BENDERS-1 phase involves what is called "complicating constraints". Nevertheless, if we decide to solve this problem no in its primal form but in its dual, the constraints are transformed into "complicating variables", this giving rise to a third Benders' algorithm (here called BENDERS-2). Fortunately, the **STEP 1** of this BENDERS-2 phase is made of separable variables, since the structure of the corresponding matrices A_1 and A_2 are, after some permutation of some rows and columns, clearly block diagonal as illustrated by Figure-2.

The **STEP 1** of BENDERS-2 phase gives rise to what we called OPTIM-0, where nV optimization problems must be solved.

A careful attention to those block-diagonal matrices shown in Figure 2 do reveal that each one of these problems can be further splitted in two others, since the block-diagonal matrices are also block-diagonal. The first one related to block k involves $nV + 3.\#\delta^+(k)$ decision variables while the second one involves $2.\#\delta^+(k)$ decision variables. In the example mentioned in the Introduction, where nV = 60 and nE = 80, the quantity $nV + 3.\#\delta^+(k)$ typically is 65 while the quantity $2.\#\delta^+(k)$ is typically 10, values significantly smaller than the original 17021.

A_1^T	B_1^T	0								
0	C_1^T	0								
0	0	F_1^T								
			A_2^T	B_2^T	0					
			0	C_2^T	0					
			0	0	F_2^T					
		...						...		
		...						...		
								A_{nV}^T	B_{nV}^T	0
								0	C_{nV}^T	0
								0	0	F_{nV}^T

Figure-2. Constraint matrix at **STEP 1** of BENDERS-2 phase.

The interaction of all these Benders' algorithms can be visualized in Figure-3 where **STEP 1** of BENDERS-*i* algorithm gives rise to the BENDERS-*(i + 1)* algorithm.

There is still one problem that must be here addressed for the sake of the full understanding of the proposed method. It is related to **STEP 1** of Benders' algorithm.

The original formulation of this subproblem can be observed in the Appendix where the dual variable $\underline{\lambda}$ need to be calculated in order to be used in **STEP 3**.

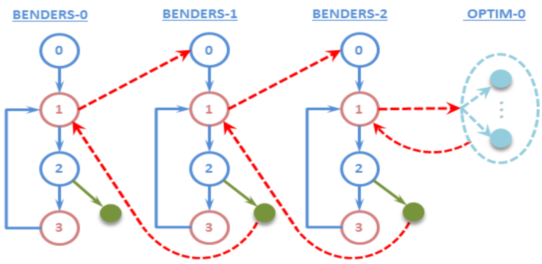

Figure-3. A chain of Benders' decomposition methods.

Again, this original formulation is not used because technically the complicating variable is fixed but it is not removed from the formulation. Its removal is elementary but in this new formulation, we loose the capacity of determining the dual variable $\underline{\lambda}$.

This limitation can be circumvented if we take into account that the optimal solution for both problems is the same and known then the Karush-Kuhn-Tucker conditions can be used in order to access $\underline{\lambda}$.

CONCLUSIONS

This paper proposes a decomposition procedure to deal with the challenging problem of energy-aware IP Traffic Engineering (TE) that complies with OSPF protocol and ECMP rule. The presented method focus on solving a large-scale TE problem that optimally finds the links and nodes that must be turned-off in order to save the maximum amount of allowed energy.

Since the literature reveals that this problem, when applied to practical cases is highly demanding as far as computational resources is concerned, most of the researchers have directed their efforts towards alternative heuristic procedures in order to alleviate this computacional burden.

In order to illustrate this fact, an important work presented by Gianoli (Amaldi et al., 2013), described an alternative method to solve the problem here discussed motivated by the fact that solutions of the original MILP formulation are extremely time consuming and capable of draining all computer resources such as memory. He used in his article network instances from the SNDlib (Orlowski *et al*., 2007) and in some cases it was impossible to find a result due to out of memory. For network Diyuan-30, for example, with 11 nodes and 42 links, solution could not be reached after more than 20 hours.

Although decomposition techniques are available for more than four decades, to the best of the authors' knowledge, the approach here presented, whereby several decompositions are used in cascade, is original.

It should be pointed that proving the efficiency of the proposed method dispenses with numerical results, since we have demonstrated the feasibility of splitting the original linear problem into several smaller ones, also linear, but of infinitely lower order, whose solution is undeniably possible at reasonable times.

However, for future works we consider that it would be interesting applying the proposed technique to various existing networks extracted from SNDlib in order to make a comparative analysis with heuristic methods reported in the literature. In addition, we suggest applying the technique to solve similar problems, considering even relaxing some restrictions, such as ECMP load balancing, for example.

APPENDIX

The Benders' algorithm applied to the split problem previously mentioned can be described by the following steps:

STEP 0: Initialization

$$v = 1$$

$$\left.\begin{array}{l} \min_{\underline{x},\sigma} \; z = \underline{a}^T \cdot \underline{x} + \sigma \\ \underline{x}^{down} \le \underline{x} \le \underline{x}^{up} \\ \sigma \ge \sigma^{down} \end{array}\right\} \Rightarrow \underline{x}^{(v)},\ \sigma^{(v)}$$

STEP 1: Subproblem Solution

$$\left.\begin{array}{l} \min_{\underline{y}} \; z = \underline{b}^T \cdot \underline{y} \\ \text{subject to:} \\ \boldsymbol{C} \cdot \underline{x} + \boldsymbol{D} \cdot \underline{y} \le \underline{s} \\ \underline{x} = \underline{x}^{(v)} : \underline{\lambda} \\ \underline{0} \le \underline{y} \le \underline{y}^{up} \end{array}\right\} \Rightarrow \underline{y}^{(v)},\ \underline{\lambda}^{(v)}$$

STEP 2: Convergence Checking

$$z_{up}^{(v)} = \underline{a}^T \cdot \underline{x}^{(v)} + \underline{b}^T \cdot \underline{y}^{(v)}$$

$$z_{down}^{(v)} = \underline{a}^T \cdot \underline{x}^{(v)} + \sigma^{(v)}$$

if $z_{up}^{(v)} - z_{down}^{(v)} < \varepsilon$ then STOP. Optimal solution.

else $v = v + 1$

STEP 3: Master Problem

$$\left.\begin{array}{l} \min_{\underline{x},\sigma} \; z = \underline{c}^T \cdot \underline{x} + \sigma \\ \text{subject to:} \\ \underline{b}^T \cdot \underline{y}^{(k)} + \left[\underline{\lambda}^{(k)}\right]^T \cdot \left[\underline{x} - \underline{x}^{(k)}\right] \ge \sigma \quad k = 1, \ldots, v-1 \\ \underline{0} \le \underline{x} \le \underline{x}^{up} \\ \sigma \ge \sigma^{down} \end{array}\right\} \Rightarrow \Gamma$$

Where $\Gamma = \underline{x}^{(v)},\ \sigma^{(v)}$

Go to **STEP 1**.

REFERENCES

[1] Amaldi E., Capone A. and Gianoli L. G. 2013. Energy-aware IP traffic engineering with shortest path routing. Computer Networks, Vol. 57, No. 6, pp. 1505-1517.

[2] Antonio C., Dan K. and Zhisheng N. 2012. Editorial for Computer Networks special issue on "Green communication networks", journal homepage: www.elsevier.com/ locate/comnet, Computer Networks, Vol. 56, No. 10, pp. 2317-2319.

[3] Aruna Prem B., Claude C., Dario R. and Jean-Louis, R. 2010. A survey of green networking research, IEEE Communication surveys & tutorials.

[4] Awduche D., Chiu A., Elwalid A., Widjaja I. and Xiao X. 2002. Overview and principles of Internet traffic engineering. RFC 3272, Internet Engineering Task Force.

[5] Benders J. F. 1962. Partitioning methods for solving mixed variables programming problems, Numerische Mathematik, Vol. 4, pp. 238–252.

[6] Ben-Tal A., El Ghaoui L. and Nemirovski A. 2009. Robust optimization, Princeton University Press.

[7] Chiaraviglio L., Mellia M. and Neri F. 2009. Reducing energy consumption in backbone networks, Proceedings of IEEE ICC.

[8] Christensen K. 2003. The next frontier for communications networks: power management, Proc. SPIE - Performance and Control of Next Generation Communications Networks, pp. 1–4.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

[9] Chu H.-W., Cheung C.-C., Ho K.H. and Wang N. 2011. Green MPLS Traffic Engineering. Atnac.

[10] Conejo A. J., Castillo E., Minguez R., and Garcia-Bertrand R. 2006. Decomposition Techniques in Mathematical Programming: Engineering and Science Applications, Springer.

[11] Gupta M., and Singh S. 2003. Greening of the Internet, SIGCOMM, ACM, pp. 19-26.

[12] Iyer S., Bhattacharyya S., Taft N. and Diot C. 2003. An approach to alleviate link overload as observed on an IP backbone, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies.

[13] Koster A. M. C. A. and Muñoz X. 2009. Graphs and Algorithms in Communication Networks: Studies in Broadband, Optical, Wireless and Ad Hoc Networks, Springer-Verlag, p. 199.

[14] Lambert S., Van Heddeghem W., Vereecken W., Lannoo B., Colle D., Pickavet M. and Ferry J. 2012. Worldwide electricity consumption of communication networks. Optics express, Vol. 20, No. 26, pp. B513-B524.

[15] Lappeteläinen A. 2011. Equal Cost Multipath Routing in IP Networks, PhD Thesis, Aalto University School of Science and Technology.

[16] Mulyana E. and Killat U. 2004. Optimization of IP networks in various hybrid IGP/MPLS routing schemes, In Proceedings of the 3rd Polish-German Teletraffic Symposium PGTS, pp. 295-304, Dresden Germany.

[17] Pióro M. and D. Medhi. 2004. Routing, flow and capacity design in communication and computer networks, Morgan Kaufman.

[18] Orlowski S., Wessäly R., Pióro M. and Tomaszewski A. 2007. SNDlib 1.0 Survivable Network Design library, Proceedings of the 3rd International Network Optimization Conference (INOC 2007), Vol. 55, No. 3, pp. 276–286, http://sndlib.zib.de/home.action.

[19] Skivée F., Balon S. and Leduc G. 2006. A scalable heuristic for hybrid IGP/MPLS traffic engineering - Case study on an operational network, Networks, 2006. ICON'06. 14th IEEE International Conference on , Vol. 2.