Near optimal link on/off scheduling and weight assignment for minimizing IP network energy consumption

Po-Kai Tseng, Wei-Ho Chung *

Research Center for Information Technology Innovation, Academia Sinica, Taipei City 11529, Taiwan

1. Introduction

The rapid pace of climate change has the potential to threaten the long-term sustainability of modern living standards. As highly industrialized economies and developing economies continue to expand, enormous quantities of greenhouse gases (GHGs) are being produced and discharged into the Earth’s atmosphere. When solar radiation penetrates the atmosphere, it is absorbed by the planet and infrared radiation is emitted. Under normal conditions, the infrared radiation would dissipate into outer space; however, the GHGs in the atmosphere trap or reflect the radiation, which results in rising temperatures [1,2].

With the growing demand for data communications, large bandwidth networks are being widely adopted. Numerous types of personal, commercial, and industrial communication equipment are being deployed, and energy consumption is increasing accordingly. Even though information and communication technologies (ICT) have improved nearly every aspect of our lives, the energy consumption of ICT equipment is a major challenge. For example, as networks become larger, routers need more processing power; and, because routers operate at high speeds, heat dissipation equipment is required [3]. Such equipment increases network energy consumption further along with the cost of network operations.

A summary of the energy consumption of network devices is given in [4]. The report states that, in the United States, the energy consumption of non-residential office and telecommunications equipment was 74 TW-h in 2000. On an annual basis, that is approximately 2% of the total national electricity consumption and costs about $6 billion per year [5]. In addition, the studies in [6,7] show that Internet equipment in the US consumes 2–8% of the electrical power. The energy consumption of ICT equipment is astonishingly high. Thus, there is an urgent need to improve energy efficiency and develop green solutions for telecommunication networks.

In recent years, the energy consumption problem has generated a great deal of interest among researchers. As a result, various mechanisms that switch network equipment from active mode to sleep mode are now widely used. In [8], Yamanaka et al. proposed a link on/off selection algorithm called DAPDNA-2 for IP and Ethernet networks. The algorithm, which is an extension of the set cover problem, searches for a set of active links to reduce network energy consumption. Meanwhile, Andrews et al. [9] introduced two scheduling approaches, called schedule with coordination (SWC) and schedule without coordination (SWOC), to minimize network energy consumption. SWC applies the fairest-to-go scheduling protocol on each link of a connection to minimize the active duration. SWOC activates each link for a specific period when a packet

* Corresponding author. Tel.: +886 2 27872364; fax: +886 2 27872315.
E-mail address: whc@citi.sinica.edu.tw (W.-H. Chung).

© 2011 Elsevier B.V. All rights reserved.

arrives and uses the first-in-first-out strategy to schedule packets. Routing and link bandwidth utilization are not considered by SWC and SWOC.

Chabarek et al. [3] formulated network energy consumption as an optimization problem and tried to minimize the cost of network provisioning by considering routing and link bandwidth utilization jointly. However, because the problem is NP-hard, it is only suitable for small-scale networks. For large-scale networks, Chiaraviglio et al. [10] developed a heuristic approach to reduce the computational complexity. In addition to the above approaches, a number of works focus on the design of green wireless networks [11,12], green IP over WDM networks [13,14], and green routing protocols [15].

In an IP network operating on the OSPF protocol, each packet is forwarded via the shortest path, which is determined by the weight assigned to each link. However, existing approaches for IP networks only utilize a node or link to switch the power on/off to conserve energy. To the best of our knowledge, the approach presented in this paper is the first one that jointly considers link on/off scheduling, network connectivity, weight assignment, and link utilization to minimize energy consumption in IP networks. We formulate the problem as a mixed integer non-linear programming problem in which the network energy consumption is minimized, subject to the flow conservation and link capacity constraints.

Although the problem can be formulated entirely as an optimization problem, the large number of integer variables and constraints impose a high level of complexity on computing a globally optimal solution. Hence, we decompose the problem into two feasible sub-problems, called the Link On/Off and Connectivity (LOOC) problem, and the Weight Assignment and Capacity Constrained (WACC) problem. The goal of the LOOC problem is to select a set of links to be active, while maintaining the connectivity of the network topology; that is, there exists at least one path comprised of active links for each source-destination pair. The WACC problem computes the weight of each link such that the routing paths can be located correctly on the active links computed from LOOC and follow shortest path routing, while satisfying the constraints on link capacity.

Genetic algorithm (GA) based approaches [16–18] and simulated annealing (SA) based approaches [19,20] have been developed to address the LOOC and WACC problems respectively (referred to as GA_LOOC and SA_WACC). To approximate the optimal solution, we propose an energy efficient approach, called the Compression Algorithm (CA), which takes an upper bound and lower bound pair as input. The CA performs implements the GA_LOOC and SA_WACC repeatedly and compresses the gap between the upper bound and lower bound so as to approximate the optimal solution. The algorithm can also be extended to handle various network equipment shutdowns, such as router, base station, and switching failures.

The active links selected by the CA scheme are used to construct a network topology that guarantees packet delivery without causing traffic congestion. In the developed system, the network control and management component is responsible for constructing the network topology. When the network link state changes (i.e., from heavy traffic demand to light traffic demand, or vice versa), the network controller transmits a set of pre-determined link metrics to all routers, and simultaneously ensures that the physical capacity constraints are satisfied.

We conduct simulations on benchmark networks and compare the proposed scheme’s performance in two scenarios: (1) route construction without considering power savings; and (2) route construction using minimum power saving without link capacity constraints. In the evaluation, we apply two widely used metrics: total network energy consumption and link load distribution performance. The numerical results demonstrate that the proposed scheme achieves energy efficiency, and simultaneously maintains the link capacity constraints.

The remainder of this paper is organized as follows. In Section 2, we formulate the problem; and in Section 3, we describe the problem decomposition technique and the proposed heuristic scheme. In Section 4, we use an example to illustrate the proposed scheme. We also show that the gap between the optimal solution and that derived by the proposed heuristic is small. In Section 5, we present the simulation results and compare our scheme’s performance with that of other heuristic schemes. Section 6 contains some concluding remarks.

2. Problem formulation

We formulate the problem of minimizing IP network energy consumption with link on/off switching and weight assignment as a mixed integer non-linear programming (MINP) problem. Given a physical network topology G(N,L) and the demand volume for all source-destination (SD) pairs, the proposed approach tries to find a set of active links and determine the link weights for the routing paths, while satisfying the traffic engineering constraints. The objective is to minimize the total network energy consumption. We define the total network energy consumption as the total number of active links multiplied by their energy consumption.

Next, we define the notations and decision variables used in the remainder of the paper.

Notations

\[ \begin{align*}
L & \text{ set of network links} \\
N & \text{ set of network nodes} \\
C_l & \text{ physical capacity of link } l \\
\Psi & \text{ set of source-destination (SD) pairs} \\
P_{\Psi} & \text{ candidate path set for SD pair } \Psi \\
\tau_{\Psi} & \text{ traffic demand of SD pair } \Psi \\
\eta_l & \text{ the required utilization of link capacity } C_l, 0 \leq \eta_l \leq 1 \\
E_l & \text{ energy consumption of link } l \\
\delta_{pl} & = 1 \text{ if path } p \text{ includes link } l; \text{ and } 0 \text{ otherwise} \\
\end{align*} \]

Decision variables

\[ \begin{align*}
x_l & = 1 \text{ if link } l \text{ is active; and } 0 \text{ otherwise} \\
y_p & = 1 \text{ if path } p \text{ is used; and } 0 \text{ otherwise} \\
w_l & \text{ link weight of the shortest path} \\
F_l & \text{ traffic allocated to link } l \\
\end{align*} \]

Problem (MINP):

\[ \begin{align*}
& \text{min} \sum_{l \in L} E_l x_l \\
& \text{subject to} \quad x_l = 0 \text{ or } 1 \quad \forall l \in L \\
& \quad y_p = 0 \text{ or } 1 \quad \forall p \in P_{\Psi}, \Psi \in \Psi \\
& \quad \sum_{p \in P_{\Psi}} y_p = 1 \quad \forall \Psi \in \Psi \\
& \quad \sum_{p \in P_{\Psi}} y_p \delta_{pl} \leq x_l, \quad \forall \Psi \in \Psi, \quad l \in L \\
& \quad \sum_{p \in P_{\Psi}} \sum_{l \in L} y_p \delta_{pl} w_l \leq \sum_{l \in L} \delta_{ql} w_l \quad \forall q \in P_{\Psi}, \Psi \in \Psi, \quad q \neq p \\
& \quad \sum_{\Psi} \sum_{p \in P_{\Psi}} y_p \delta_{pl} \tau_{\Psi} = F_l \quad \forall l \in L \\
& \quad F_l \leq \eta_l C_l \quad \forall l \in L \\
& \quad w_l \in Z^+ \quad \forall l \in L \\
\end{align*} \]
i.e., \( x_i = 1 \) if link \( l \) is active; and 0 otherwise. In Constraint (2), the variable \( y_p \) is binary, i.e., \( y_p = 1 \) if path \( p \) is used; and 0 otherwise. Constraint (3) stipulates that only one candidate path can be used for each SD pair. The left-hand side of Constraint (4) is 1 if link \( l \) is active; otherwise, it is 0. Constraint (5) guarantees that, for each SD pair, the selected path \( p \) is the shortest path with respect to the link cost metric \( w_l \). In other words, the length of the selected path \( p \) must be less than or equal to the length of the candidate paths for other SD pairs. Constraint (6) computes the traffic allocated on each link. Constraint (7) guarantees that the traffic generated by each SD pair is delivered via the shortest path while satisfying the link capacity constraints. The link weight is bounded by the OSPF link weight boundary in Constraint (10).

In the following subsections, we propose heuristic schemes to solve the LOOC and WACC sub-problems.

### 3.2. Genetic heuristic for the LOOC sub-problem

To address the LOOC problem, we propose a genetic algorithm (GA), which is an evolutionary heuristic technique designed to solve complex optimization problems [16]. The algorithm is comprised of five stages: chromosome, fitness, crossover, mutation, and survivor selection. It compiles a population from a set of individuals, each of which represents a candidate solution to the target problem. The unique characteristics of an individual are used to form a chromosome, which is a string of genes. After a chromosome has been generated, a fitness function determines its goodness. For a minimization problem, an individual with a lower fitness function value is regarded as a better individual.

Next, the GA randomly initializes a population and improves it through the crossover, mutation, and survivor selection operations. The iterative process continues until a stopping criterion is satisfied. Usually, the criterion is defined as the maximum number of iterations or a threshold on the difference between the fitness function values of consecutive generations.

We now describe the genetic operators for the LOOC problem.

(a) Chromosome
In the LOOC problem, a chromosome represents all the links in a network. A gene corresponds to a link, and the value of each gene is a binary symbol. A link \( l \) is active if the value of the corresponding gene is 1; otherwise, the link is inactive. A gene's value is 1 if the corresponding link \( l \) is active; otherwise, its value is 0.

\[
g_i = \begin{cases} 1, & \text{if link } i \text{ is turned on, } \\ 0, & \text{otherwise}, \end{cases} \quad (11)
\]

where \( g \) represents the gene of link \( i \). A chromosome is represented by a string whose length is equal to the number of links in the network. An example of a chromosome is shown in Fig. 2, where \( |L| \) denotes the number of links; links 1, 3, and 4 are active; and links 2, 5, and \( L \) are inactive.

(b) Fitness function
A fitness function is used to differentiate good and bad chromosomes so that the population evolves toward the desired objective. The goal of Sub-problem 1 (LOOC) is to select the on mode or off mode for each link while maintaining at least one connected path for each SD pair. Hence, we define the fitness function for a chromosome as the number of non-connected pairs.

(c) Crossover
We randomly select two chromosomes from the population as parents and use them to generate children via the crossover operation. For each gene in one of the two chromosomes, we randomly generate a real number within the interval [0,1] and define the crossover probability \( p_c = 0.9 \). If the generated number of a gene is less than \( p_c \), the gene is exchanged for one of the genes in the other chromosomes; otherwise, no gene exchanges occur. This operation is shown in Fig. 3, where the second, fourth and fifth genes (gray boxes) are selected to perform gene exchange.
(d) Mutation
The mutation operation increases the amount of variation in the evolutionary process. For each child, we set a mutation probability $p_m = 0.1$ to determine whether a gene's binary symbol should be changed. Similarly, a real number is generated randomly within $[0,1]$ for each gene. If the generated number of a gene is less than $p_m$, the gene's binary symbol is changed by an XOR operation; otherwise, the binary symbol is not changed.

(e) Survivor selection
The crossover and mutation operations are repeated until the number of children is equal to the size of population. Then, we remove the best chromosome from the parents and children and insert it into a new population until the size of the new population equals a predefined size.

This GA-based heuristic scheme is called the GA_LinkOn/Off and Connectivity (GA_LOOC) heuristic scheme. The steps of the algorithm are listed in Fig. 1, where $X$ represents a chromosome and $x_i$ denotes the gene associated with link $i$. LinkON$_{lb}$ and LinkON$_{ub}$ are, respectively, the lower bound and upper bound on the number of active links. The non-connection ($X$) is used to compute the number of non-connected pairs based on the chromosome $X$. The initial population is generated in line 6; and a new population is generated from line 14 to line 20. The GA_LOOC heuristic scheme terminates if there exists a chromosome $X$ whose non-connection ($X$) is equal to 0 or the maximum generation (MaxGeneration) is reached. When the scheme terminates, the best chromosome $X$ and its non-connection are returned.

3.3. Simulated annealing heuristic for the WACC sub-problem

We propose a simulated annealing [19] based heuristic scheme to solve the Sub-problem 2 (WACC). Our approach, called the Simulated Annealing based WeightAssignment and CapacityConstrained (SA_WACC) heuristic scheme, exploits the concept of simulated annealing to repeatedly tune the link metric so that the link capacity constraint can be satisfied.

The SA_WACC scheme is described in detail in Fig. 4. The simulated annealing operation is located in the nested loops from line 23 to line 52. The outer do loop controls the value of the temperature $T$, and the inner while loop repeatedly selects a link and tunes its weight so that the capacity constraint can be satisfied. Initially, SA_WACC sets $A_0$ according to the function $X(x_i)$, which extracts the value of the gene associated with link $i$ from the chromosome $X$. If $X(x_i) = 1$, the weight of link $i$ is set as $x$; otherwise, the weight is set as $1$. If $LoadRatio_{max}$ is less than or equal to the maximum acceptable $LoadRatio_{max}$, it is assigned to $LoadRatio_{max}$, and the algorithm terminates. Otherwise, the difference between $LoadRatio_{max}$ and $LoadRatio_{max - 1}$ is
assigned to Δ. If Δ is less than or equal to 0 and LoadRatio_{max,k} is less than LoadRatio_{max,k} then LoadRatio_{max,k} and A_k are assigned to LoadRatio_{max} and A respectively. Otherwise, the exponential of −εΔ/T is computed and a random real number within (0,1) is generated. If the exponential of −εΔ/T is less than the generated number, the best chromosome X_opt, non-connection_{opt}, A_{opt}, LoadRatio_{opt} are assigned to X, non-connection, A, LoadRatio_{max} respectively.

When the proposed SA_WACC scheme terminates, we obtain the current best link weight (A) and the maximum acceptable FlCl ratio LoadRatio/C3max.

3.4. Compression Algorithm (CA)

Recall that the objective of MINP is to compute the minimum network energy consumption. We assume the energy consumption of all links is equal. Then, the objective function can be rewritten as the equivalent equation \( P = \frac{1}{2} L x \).

We use the proposed Compression Algorithm (CA) to find the minimum number of active links in a network under Constraints (1)–(10). More specifically, the algorithm combines the GA_LOOC and SA_WACC heuristic schemes and repeatedly compresses the gap between LinkONub and LinkONlb to determine the minimum number of active links. Fig. 5 provides a complete description of the CA algorithm.

The main operation of the CA scheme is located in the while loop (lines 12–33), which compresses the gap between LinkONub and LinkONlb repeatedly until it is less than the acceptable gap (\( h \)). First, we update the best chromosome X_opt, non-connection_{opt}, A_{opt}, LoadRatio_{opt} by X, non-connection, A, LoadRatio_{max} respectively. If count is less than MAXCount, it is increased by 1; otherwise, \( P_l x(l) \) is assigned to LinkONlb and count is set at 1.

Fig. 5. Pseudo code for the CA.
The while loop terminates when the gap between \( \text{LinkON}_{ub} \) and \( \text{LinkON}_{lb} \) is less than or equal to the threshold gap (\( \eta \)). When the CA terminates, we obtain the current best chromosome \( \mathbf{X}^{\text{opt}} \) (for the on/off links), the best link weight \( \mathbf{A}^{\text{opt}} \) (for routing), and the minimum network energy consumption.

4. Case study

We conducted simulations to evaluate the performance of the proposed approach. In Fig. 6, we provide illustrative examples of the MINP model and the CA scheme. Fig. 6(a) shows the test topology. The traffic demand of each pair of nodes is 5 megabytes (MB), the capacity of each link is 50 MB, and \( \eta_l = 0.7 \) for each link \( l \). The convergence process of CA is shown in Fig. 6(b). The x-axis represents the cumulative computation time from iteration 1 to iteration 6. The CA was terminated after 33.703 s when \( \text{LinkON}_{ub} = \text{LinkON}_{lb} \) in iteration 6. Fig. 6(c) and (d) show the results derived by the MINP model and the CA respectively. Each link is labeled with a pair of numbers, (weight, load). The link weight was set at 10,000 for CA if the link was turned off. We used LINGO optimization software to solve the MINP model. This simulation was executed on a PC running Windows XP with a 2.53 GHz CPU.

The thick and thin arrows represent, respectively, the on status and off status of the links. We observe that the load on each link is less than or equal to 35 (satisfying Constraint (7)), and each pair is connected by links that are switched on. Moreover, based on the computed link weights, all the routing paths can be located on the shortest paths correctly. In other words, each routing path can be routed via the active links on the shortest path. It is noteworthy that the proposed CA heuristic has the same number of active links (i.e., the same network energy consumption) as the optimal solution derived by the MINP model.

5. Simulation results

We implemented the CA scheme on the COST239, GTE and NSF benchmark networks shown in Fig. 7. In each network, two adjacent nodes are connected by two opposite directional links. In the simulations, we compared the performance of three schemes: the proposed CA, route construction with minimum–maximum link load without considering power savings (No Power Saving), and route construction using minimum power saving without considering link capacity constraint (No Cap. Const.). The simulation program was written in C programming language and operated on a PC running Windows XP, with a 2.53 GHz CPU and 2 GB RAM.

The simulations are divided into two parts. (1) We use a fixed link capacity, uniform pair traffic demands and various link utilization requirements (i.e., \( \eta_l \) used in the MINP model) to compare the network energy consumption and the maximum link utilization ratio (=max\{\( F_l/C_l \)/\( l \in L \)) under benchmark networks running the CA, No Power Saving and No Cap. Const. schemes. (2) We partition the NSF network into five time zones and use a fixed link capacity, fixed link utilization requirement, and non-uniform pair traffic demands according to different time zones. Then, we observe the variation in the network energy consumption and maximum link utilization ratio under the NSF network running CA, No Power Saving and No Cap. Const. schemes.
For the No Power Saving and No Cap. Const. schemes, we formulate two integer linear programming models and solve them directly with LINGO optimization software. The optimization formulas for the two models are as follows.

**Problem (No Power Saving):**

\[
\begin{align*}
\text{min} & \quad u \\
\text{subject to} & \quad \sum_{l \in L} x^l_t - \sum_{l \in L} x^l_{t-1} = \begin{cases} 
-1, & \text{if } n = \psi^\text{src} \\
0, & \text{if } n \neq \psi^\text{src}, \psi^\text{dest} \forall n \in N, \psi \in \Psi \\
1, & \text{if } n = \psi^\text{dest} 
\end{cases} \\
\sum_{\psi \in \Psi} x^l_{t-1} \psi = F_l & \forall l \in L \\
x^l_t = 0 & \text{or } 1 \quad \forall l \in L, \quad \psi \in \Psi 
\end{align*}
\]

The objective of No Power Saving is to minimize the load on the most congested link. Constraint (12) guarantees that there is a path for each SD pair, where \(\psi^\text{src}\) is the source of pair \(\psi\), and \(\psi^\text{dest}\) is the destination of pair \(\psi\). Constraint (14) is used to find the load of the most congested link. In Constraint (15), \(x^l_t\) is a binary variable. If pair \(\psi\) uses link \(l\) to deliver a traffic demand, \(x^l_t = 1\); otherwise, \(x^l_t = 0\).

**Problem (No Cap. Const.):**

\[
\begin{align*}
\text{min} & \quad \sum_{l \in L} E_l v_l \\
\text{subject to} & \quad \sum_{l \in L} x^l_t + t^l_d = \sum_{l \in L} x^l_{t-1} \forall n \in N, \ d \in N, \ d \neq n \\
\sum_{l \in L} x^l_d = \sum_{m \in N(d)} t^l_d & \forall d \in N \\
\sum_{l \in L} x^l_d = 1 & \forall n \in N, \ d \in N, \ d \neq n \\
x^l_t \leq M x^l_d & \forall l \in L, \ d \in N \\
\sum_{d \in N} x^l_d \leq v_l & \forall l \in L \\
x^l_t = 0 & \text{or } 1 \quad \forall l \in L, \ d \in N 
\end{align*}
\]

The objective of No Cap. Const. is to minimize the network energy consumption. Constraints (16) and (17) guarantee the law of flow conservation. Constraint (18) requires each node to turn on an output link for each destination. Constraint (19) stipulates that link \(l\) can only carry packets to destination \(d\) if the variable \(x^l_t\) is 1, where \(M\) is a large enough number. Constraint (21) is to determine the value of \(v_l\). If \(x^l_t = 1, v_l = 1\). Constraint (22) requires that variable \(x^l_t\) is binary. If link \(l\) is used to carry packets to destination \(d\), \(x^l_t = 1\); otherwise, \(x^l_t = 0\).

We implement the three compared schemes on each benchmark network. The input parameters for CA are MaxCount = 50, \(\text{LinkON}_{\text{src}} = |L|\), \(\text{LinkON}_{\text{dest}} = |N|\), \(\theta = 1\), \(T = 10\), \(\epsilon = 0.8\), \(\alpha = 100\), \(\rho = 8\), \(|\chi| = 2000\), MaxGeneration = 300, \(p_c = 0.9\), and \(p_m = 0.1\). The capacity of each link is set at 500 MB and a link’s energy consumption is set at 1000 watts for each benchmark network.

5.1 Uniform traffic demand

We compare the energy consumption and maximum link utilization ratio of the benchmark schemes. The results are shown in Fig. 8(a) and (b) respectively. In each sub-figure, the x-axis represents the SD pair traffic demand in megabytes (MB), and the y-axis represents the energy consumption in watts. The SD pair traffic demand is 5 MB, which means that each SD pair has the same traffic demand of 5 MB. In the figure legends, CA-0.9, CA-0.8 and CA-0.7 mean that we implement the CA scheme with \(\eta_l = 0.9, 0.9, 0.9\) and \(0.7\) respectively for each link \(l\). The No Cap. Const. and No Power Saving do not have a link utilization limit.

In Fig. 8(a), the No Cap. Const. scheme achieves its objective, i.e., it minimizes the energy consumption. Since the No Power Saving scheme tries to minimize the load on the most congested link, it must turn on all the links to achieve its goal; hence, the scheme has the highest energy consumption. Under the CA scheme, the energy consumption increases with the SD pair traffic demands for each benchmark network. We also find that CA can reduce the energy consumption if the network has a higher node degree (number of links/number of nodes).

In Fig. 8(b), the maximum link utilization ratio increases with the SD pair traffic demands for the No Cap. Const. and No Power Saving schemes in each benchmark network. The No Cap. Const. scheme always has the highest link utilization ratio and the No power Saving scheme always has the lowest link utilization ratio because of their respective objective functions. For CA, the maximum link utilization ratio decreases with the value of \(\eta_l\). In addition, we observe that all the link utilization requirements \(\eta_l\) are satisfied.

Even though No Cap. Const. saves the most energy, in most cases, the maximum link utilization ratio exceeds 1 (i.e., most links are overloaded); while No Power Saving activates all the links to achieve the maximum link load. The trade off between the network energy consumption and the maximum link utilization ratio can be observed in Fig. 8. Our CA scheme reduces the network energy consumption under the No Power Saving scheme by 35–75%, and it satisfies the link utilization requirement \(\eta_l\).

5.2 Non-uniform traffic demands

In this simulation, we implement the CA, No Power Saving and No Cap. Const. schemes on the NSF network to observe the energy consumption and maximum link utilization ratio under non-uniform SD pair traffic demands. The link capacity is set at 1000 MB and \(\eta_l = 0.9\) for each link \(l\).

The NSF network is divided into five time zones: A, B, C, D, and E. The time difference between any two adjacent time zones is one hour, as shown in Fig. 9(a). We also divide a 24-h period into 5 slots: (1) 00:00–04:00, (2) 05:00–09:00, (3) 10:00–14:00, (4) 15:00–19:00, and (5) 20:00–24:00. Slot (1) begins at 00:00 in time zone A, 01:00 in time zone B, 02:00 in time zone C, 03:00 in time zone D, and 04:00 in time zone E. The time slots and output demands are shown in Fig. 9(b). For example, in the third row and second column of Fig. 9(b), the output demand 10 means that each router has the same output demand of 10 MB in time zone A (i.e., routers 1, 2 and 3 have the same output demand of 10 MB (see Fig. 9(a))) at midnight.

The simulation results are shown in Fig. 10. Fig. 10(a) shows that the No Cap. Const. and No Power Saving schemes achieve the maximum and minimum energy consumption respectively. Under the CA scheme, the highest energy consumption occurs in the 10:00–14:00 time slot, and the lowest in the 00:00–04:00 time slot. This observation is consistent with the fact that the output demand is the highest during the 10:00–14:00 slot, and the lowest during the 00:00–04:00 slot. Under each scheme, the maximum link utilization ratio varies with changes in the total output demand. The proposed CA scheme also satisfies the link utilization requirement \(\eta_l = 0.9\) in all time slots and consumes 40–60% less energy than the No Power Saving scheme.

6. Conclusion

We have formulated the IP network energy consumption problem as a mixed integer non-linear programming (MINP) problem...
and proposed a low-complexity scheme, called the Compression Algorithm (CA), to approximate the optimal solution of MINP. The proposed scheme uses a pair of lower and upper bounds to limit the area of feasible solutions. In each iteration, the scheme exploits the genetic algorithm technique to determine a set of active links. It then uses the concept of simulated annealing to (1) determine the weight of the active links so that the traffic flows can be delivered via the links on the shortest path; and (2) update the values of the lower and upper bounds to compress the area of feasible solutions. The CA scheme yields a near optimal solution with low complexity. Moreover, it can be extended to handle various network equipment shutdown scenarios, such as router, base station, and switching failures. In a series of simulations, we compare the performance of CA with that of a route construction scheme without considering power savings and a route construction scheme that uses minimum power saving without considering the link capacity constraints. The results demonstrate that the CA scheme can reduce network energy consumption by 35–75%, while maintaining the constraints on the link utilization requirement.

Fig. 8. Performance comparison of the benchmark networks.
References


