Topology Design of Hierarchical Hybrid Fiber-VDSL Access Networks with Enhanced Discrete Binary PSO

Rong Zhao[1,2], Yi Zhang[1], and Ralf Lehnert[1]

[1] Technische Universität Dresden, Chair for Telecommunications,
01062 Dresden, Germany
{rong.zhao,ralf.lehnert}@tu-dresden.de
[2] Detecon International GmbH, Oberkasseler Straße 2,
53227 Bonn, Germany

Abstract. As one of the most efficient access solutions, VDSL technology is becoming a highlight in the next generation network. This paper addresses the topology design of hierarchical Hybrid Fiber-VDSL Access Networks (HFVAN) as a NP-hard problem. An efficient strategy with general binary models is proposed to find a cost-effective and high-reliable network with heuristic algorithms in a short time. An enhanced Discrete Binary Particle Swarm Optimization (DBPSO) is developed and successfully implemented for this network planning problem, both for clustering and positioning. In terms of numerical results, the performance of the enhanced DBPSO is compared with some previous approaches.

Keywords: Hybrid Fiber-VDSL Access Network, network architecture, network survivability, network planning and optimization, Discrete Binary Particle Swarm Optimization.

1 Introduction

In the last decade DSL Access Network has become one of the most efficient access technologies to provide a high bandwidth for SOHO subscribers. As the milestone for DSL technology, the DSL specification VDSL2 (ITU-T, G.993.2) enables network operators to provide up to 200 Mbps bandwidth over twisted pair. This increase has a huge impact on global connectivity coupled with multifarious applications, such as IPTV in HDTV quality.

Similar to a PSTN access network, the Hybrid Fiber-VDSL access network (HFVAN) is divided into two parts: the distribution network between Central Office(CO) and Street Cabinets (SCs), and the *Last 100 Meters* network. The *Last 100 Meters* network is the part between SCs and the subscribers, where the VDSL technology is implemented to realize a high transmission speed of subscribers over twisted pair. This work focuses on the planning and optimization of the distribution network in HFVANs, i.e. finding out a low-cost hierarchical structure with suitable positions for the intermediate layer and optimal

C. Wang (Ed.): AccessNets, LNICST 6, pp. 208–223, 2009.

connections of all network nodes subject to delay and network reliability with node-biconnectivity. The hierarchical HFVAN design is NP-hard [1][2][3].

Some prior approaches addressed the topological design for hierarchical backbone networks by [4][5][6][7] and access networks by [8][9][10][11]. Generally, network structures are illustrated by hierarchical star-star, tree-star, or mesh-star topology. Most of these optimization problems are NP-hard. Due to their complexity, different methodologies are investigated, such as Linear Programming, Simulated Annealing, Tabu Search, Genetic Algorithms, Ant Colony Optimization, etc. Some algorithms have been investigated for HFV access network planning problems in [1][2][3].

Particle Swarm Optimization (PSO) is a metaheuristics, which is inspired by the observation of the behaviors of birds and fishes. This optimization method introduced by Kennedy and Eberhart in 1995, has shown an excellent performance in many applications. PSO was designed for the optimization of continuous optimization problems, in particular to some mathematical problems [12][13][14][15]. In the last years more real-world problems have been studied with PSO [16], such as constrained optimization problems [17] and the neighbor selection in Peer-to-Peer Networks [18]. Moreover, PSO-algorithms have been extended to cover multifarious discrete optimization problems [19][20][21][22]. However, only few approaches to hierarchical topology design problems with Discrete Binary PSO (DBPSO) were published in the last years. PSO with a Non-binary Model (NBM) was studied for HFVAN topology design problems, but its performance was not stable enough due to the complex encoding process [2]. This work adopts DBPSO and proposes an enhanced update function for the topology design of hierarchical HFVANs.

This paper is organized as follows: section 2 describes the hierarchical structure of HFVANs and makes a problem statement for their planning. Section 3 presents the complete strategy to fulfill the hierarchical topology design of HFVANs and binary models for its network topology. Section 4 introduces Discrete Binary Particle Swarm Optimization and proposes the enhanced DBPSO for the HFVAN topology design. Consequently, some optimization results from DBPSO will be analyzed and compared with Simulated Annealing (SA), Tabu Search (TS), Genetic Algorithms (GA), Ant Colony Optimization (ACO) and PSO with NBM in Section 5. Finally the conclusion and further work will be presented.

2 Problem Statement

2.1 Hybrid Fiber-VDSL Access Networks

In Fig.1 the Central Office (CO), Street Cabinets (SCs) and end users compose a hierarchical infrastructure of a HFVAN. Its two main parts are the distribution network (CO-SCs) and the *Last 100 meters* network (SCs-VDSL end users), which are interconnected by the optical fiber and the existing twisted pair, respectively. This work focuses on planning and optimization of the distribution network due to the fixed connection within the *Last 100 Meters* area. To reduce the investment costs and improve network reliability, the hierarchical structure

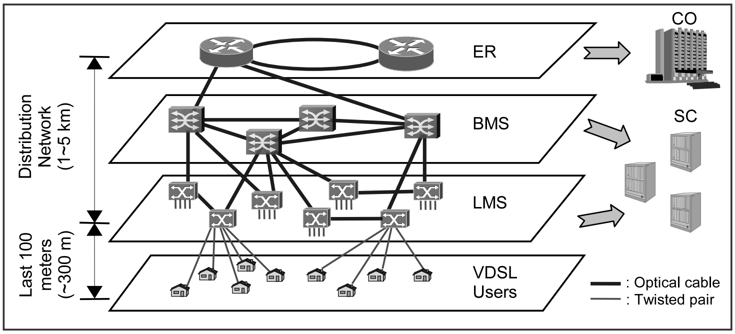


Fig. 1. Hierarchical structure of hybrid fiber-VDSL access networks

(middle layer with meshing) in the distribution network has been proposed subject to constraints of network elements, such as Edge Router (ER), Branch Micro Switch (BMS) and Leaf Micro Switch (LMS). Keeping up the infrastructure of the PSTN access network, the HFVAN respectively installs ER into CO and BMS or LMS into SC, as shown in Fig.1. All VDSL subscribers are connected with a star topology to a LMS located in the SC.

The top layer of this network is the Edge Router which provides the interface between the distribution network and the backbone network. As assumed, only one ER will be placed in this layer. Some BMSs compose the intermediate layer of this network. The third layer of the network consists of a number of LMSs, accessing to residential and SOHO subscribers. In consideration of the network availability and reliability, a LMS must have at least two paths to reach the ER, i.e. each LMS at the low layer must have two uplinks either to BMSs in the high layer or to be joined with other LMSs. The hierarchical network topology of HFVANs should fulfill the requirements of both edge-biconnectivity and node biconnectivity.

2.2 Objective Function and Constraints

The major planning problem in this work is to build up the intermediate BMS-layer of the hierarchical network. More precisely, it is to determine the corresponding number of BMSs, the position of each BMS and to find out the optimal links within the network nodes, subject to minimal costs and some constraints. The objective function (costs function) Z_{net} is formulated by:

$$Z_{net} = \sum_{i} x_i^{bms} C_{bms} + \sum_{i} \sum_{j} x_{ij} c(l_{ij}) + N_{lms} C_{lms} \tag{1}$$

where N_{lms} is the number of LMSs, C_{bms} and C_{bms} are the costs of each LMS and BMS, l_{ij} is the edge length between node i and j $(i, j \in 1..N_{lms})$, x_i^{bms} and x_{ij} are binary variables (if a node or link is selected, then 1; otherwise 0). $c(l_{ij})$ is the cost of link ij that is assumed to be a linear function of the link length l_{ij}:

$$c(l_{ij}) = C_l l_{ij} + c_{ij}^k \tag{2}$$

where C_l is the cost unit of the fiber. c_{ij}^k is the fixed cost for different link types within ER, BMSs and LMSs, i.e. k=1 for link ER-BMS, k=2 for link BMS-BMS, k=3 for link BMS-LMS. As we assume that the number and costs of LMSs and ER are fixed, therefore these costs can be ignored in the objective function.

The constraints for the planning of HFVANs are made up of *physical constraints*, *flow constraints*, maximal *delay*, and *reliability*. *Physical constraints* are related to topology, location and selection of BMSs in the HFVAN. The maximal and minimal uplink/downlink ports are limited for node degrees. Considering the characteristics of HFVANs, the network can be modeled with M/M/1 system to derive the *end-to-end delay* [1]. To deal with single link failures, a minimum connectivity of the topology should be satisfied to guarantee the network reliability. In this work edge- and node-biconnectivity for single node failures is considered.

3 Analysis of HFV Access Network Topology Design

3.1 Strategy

A complete strategy of the HFVAN topology design with heuristic algorithms is proposed in [3].

```
Initialization (network structure, algorithms parameters)
while stop_criterion=false do
    Generation of new network structures:
        Positioning of BMSs;
        Clustering of LMSs with BMSs;
    Reparation of links (BMS-LMS, LMS-LMS);
    Multi-constrained MST (ER-BMS layer);
    Augmentation (ER-BMS layer);
    Costs evaluation (total network costs);
endwhile
Output the best solution found so far.
```

Fig. 2. Strategy of the HFVAN topology design

Fig.2 presents an iterative procedure of planning the HFVAN structure with heuristic algorithms. The whole optimization process updates network structures dynamically by generating new neighbors. Firstly, the initial network structure is randomly generated, but it is ensured to be valid using some repair functions. As described in section 2, HFVANs are built with a hierarchical topology. The middle layer of BMSs, i.e. the positions and numbers of BMSs, are unknown. After positioning of BMSs and clustering of LMSs, the connections between BMSs and LMSs can be determined. With local repair functions the redundant or invalid links will be removed, a few improved links will be added, respectively. In this case a Minimum Spanning Tree (MST) is an efficient solution to build up a fundamental structure of the BMS-layer. However, multi-constraints, such as node degree, delay, capacity, make the MST problem more complex. This problem is called Multi-Constrained Minimum Spanning Tree (MCMST). Based on MCMST and Augmentation, a bi-connected, cost-effective, and degree/capacity/delay-valid network structure with three layers will be achieved. Costs evaluation will estimate network costs including the costs of LMSs, BMSs and the selected links within them. The stop criterion is assumed as a certain iteration subject to the visited solutions.

Optimization algorithms, such as DBPSO, will have significant effects on this network planning procedure, particularly for the generation of new neighbors (Positioning of BMSs and Clustering of LMSs) and Costs evaluation. In a sense other functions in the loop are necessary for HFVAN topology design, but have little influence on heuristic optimization algorithms studied. The optimal solutions found so far have less than 2% difference to reference global optima. It verifies the proposed strategy to be suitable for this design problem.

3.2 Binary Modeling

Besides the Linear 0-1 Integer Programming, the binary modeling can be applied to optimization problems with heuristic algorithms, such as Simulated Annealing, Tabu Search, Genetic Algorithms, etc. Similarly, the network topology of a HFVAN can be modeled as a particle with a group of binary individuals. Hereby, two scenarios are proposed in Fig.3. *Full Particle Modeling*(FPM) is used for DBPSO to fulfill both Positioning and Linking of BMSs, and Clustering of LMSs, including all binary codes. All changes of binary codes are dependent on DBPSO. *Partial Particle Modeling*(PPM) is adopted to present DBPSO influencing only the Positioning and Linking of BMSs, i.e. BMSs, ER-BMS, BMS-BMS in the first terms. As assumed, the links between ER and LMS are not allowed due to the high cost. In the current approach there is only one ER. Hence, the first binary code for ER should be 1.

As shown above for *Full Particle Modeling*, the overall particle length with N_{lms} SC (LMS) and 1 CO (ER) consists of different individuals numbers:

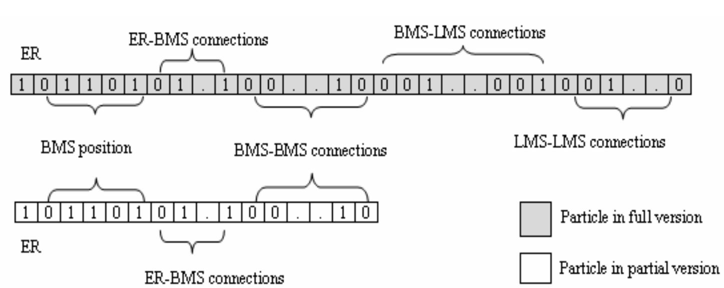


Fig. 3. Full Particle Modeling and Partial Particle Modeling for HFVANs

- first entry for ER;
- N_{lms} entries for potential BMSs' positions;
- N_{lms} entries for the connections between ER and BMSs;
- $N_{lms}(N_{lms}-1)/2$ entries for the connections within BMSs;
- N_{lms}^2 entries for the connections between BMSs and LMSs;
- $N_{lms}(N_{lms}-1)/2$ entries for the connections within LMSs;

where 1 means the presence of nodes or edges in the network solution, 0 means the absence of nodes or edges. If the first code of ER is ignored, the particle length of *Full Particle Modeling* (FPM) is described as:

$$L_{FPM} = (2N_{lms} + 1)N_{lms} \tag{3}$$

Comparatively, the particle length of *Partial Particle Modeling* (PPM) is:

$$L_{PPM} = N_{lms}(N_{lms} + 3)/2 \tag{4}$$

The length of PPM is obviously less than FPM, where a deterministic method will fulfill the connection within LMSs and BMSs. If N_{lms} is large enough, then L_{FPM} is approximated to $4L_{PPM}$. With the deterministic method the PPM could make optimization results locally optimal, but the computation time can be significantly reduced. In the previous study a non-binary model was proposed for continuous PSO [2].

4 Discrete Binary Particle Swarm Optimization

4.1 Introduction

As a swarm intelligent computation technique, PSO has its roots in the simulation of a simplified social system such as bird flocking, fish schooling, and swarm theory in particular. Different from GA and some other Evolutionary Computational (EC) algorithms, each individual in PSO searches for the optimum under the spirit of *Cooperation*.

A general strategy of PSO is described in Fig.4, which is suitable for all continuous and discrete PSO algorithms. The main procedure of PSO is fulfilled by the

```
Initialization (position, velocity, PSO parameters)
while stop_criterion=false do
   Schedule_Activities
      Evaluate positions (calculate solutions);
      Find global optimum and personal optimum;
      Update velocity and position;
   end Schedule_Activities
endwhile
Output the optimal solution.
```

Fig. 4. Pseudo code of Particle Swarm Optimization

Schedule-Activities: calculate solutions, find the current best value of all particles and the personal best value of each particle, update the velocity and position.

The intensification of an optimization process with PSO is fulfilled by keeping or strengthening a useful particle, i.e. the particle with the best value found so far. It can accelerate the convergence of the search processes. Inversely, the personal optimum and the random change can improve the diversification of PSO to avoid the local optimum.

4.2 Formulation and Notation

The standard PSO which operates in a continuous search space is suited to handle real valued optimization problems. But many optimization problems are set in a discrete search space. The algorithm tends to fall apart if a particle is flying to either zero and one. A discrete binary version of the PSO was introduced by Kennedy and Eberhart [19], using the concept of velocity as a probability that a bit takes on 0 or 1. The original update functions for the velocity and the position of particles in DBPSO are described as:

$$v_n(k+1) = v_n(k) + c_1 r_1 (p_{best,n} - x_n(k)) + c_2 r_2 (g_{best} - x_n(k)) \tag{5}$$

$$x_n(k+1) = \begin{cases} 0 & sign(v_n(k+1)) \leq r_3 \\ 1 & sign(v_n(k+1)) > r_3 \end{cases} \tag{6}$$

$$sign(\alpha) = 1/(1+e^{-\alpha}) \tag{7}$$

where
c_1 and c_2 are the acceleration factors with constant and positive values;
V_{max} is the maximum of the velocity;
N_p is the number of particles;
n is the n-th particle in the swarm, $n \in [1, N_p]$;
k is the iteration number;
$v_n(k)$ is the velocity of the particle n, $|v_n(k)| \leq V_{max}$;
$x_n(k)$ is the position of particle n, binary variable;
$p_{best,n}$ is the best position of particle n;
g_{best} is the best position for all particles;
r_1, r_2, r_3 are uniformly distributed number between 0 and 1.

4.3 Enhanced DBPSO

Obviously, the acceleration factors c_1 and c_2 are two important parameters influencing the performance of DBPSO. c_1 can strengthen the effect of $p_{best,n}$, c_2 emphasizes the effect of g_{best}, respectively. They control each optimization step, in which a particle will move to either the best personal position $p_{best,n}$ or the best global position g_{best}. A suitable setting of two parameters can lead particles to reach the balance between *Cognition* and *Social* behaviors well [13]. Anyway, the independence of c_1 and c_2 makes it difficult to apply DBPSO for real-world problems.

Generally, V_{max} in DBPSO is taken as a complementary factor to limit the further exploration rate. The velocity should be limited to V_{max}. A high V_{max} in the continuous-valued version increases the range explored by a particle. Conversely, a smaller V_{max} leads to a higher mutation rate [19]. However, numerical results in this work show that V_{max} plays an important role in the HFVAN topology design by updating the velocity and position. An inefficient setting of V_{max} could disturb the performance of DBPSO. Some numerical results will be shown later.

To avoid this negative effect, an enhanced update function is proposed, where V_{max} is taken into account. However, adding a new parameter into the update function could make it more difficult to obtain a suitable configuration for all control parameters. To reduce the complexity, we will combine a few control parameters in the current update function of DBPSO.

In the DBPSO $|r_1(p_{best,n} - x_n(k))|$ and $|r_2(g_{best} - x_n(k))|$ are less than 1. Actually, the main change of $v_n(k+1)$ strongly depends on the ratio of c_1 and c_2 besides the prior state $v_n(k)$. We assume $c_1 + c_2 = V_{max}$. Then the ratio $\gamma = c_2/c_1$ is defined to describe the effects of $p_{best,n}$ and g_{best}. Therefore, a new update DBPSO function is proposed as:

$$v_n(k+1) = v_n(k) + V_{max}\frac{1}{1+\gamma}r_1(p_{best,n} - x_n(k)) + V_{max}\frac{\gamma}{1+\gamma}r_2(g_{best} - x_n(k)) \tag{8}$$

where the old control parameters c_1, c_2, V_{max} are replaced by γ and V_{max}. The further characteristics of γ and the performance of the enhanced update function will be discussed later in terms of numerical results.

4.4 Application

Tab.1 provides a mapping from DBPSO to HFVAN topology design problems. Other parameters have been explained in *Formulation and Notation*.

5 Results and Analysis

5.1 Optimization Environment

1) Test bed: three typical networks are studied.

- Network I: 13 Street Cabinets (SCs) and 1 Central Office (CO), where CO is located at the boundary of the network;
- Network II: 40 SC and 1 CO , the position of CO is similar to that in Network I;
- Network III: 87 SC and 1 CO, where CO is located in the middle of the network.

Table 1. Mapping of DBPSO in HFVANs topology design

DBPSO	Application
Particle n	n-th network topology
Code i in particle n	i-th node (or edge) in n-th network topology
Modification of particles	Generation of new topologies
Position $x_{n,i}(k) = 0\ or 1$	State of i-th node (or edge) in n-th topology (1:selected, 0:unselected)
Velocity $v_{n,i}(k) = 0\ or 1$	State change of i-th node (or edge) in n-th topology
Personal best position $p_{best,n,i} = 0\ or 1$	State of i-th node (or edge) in the local best n-th topology
Global best position $g_{best,i} = 0\ or 1$	State of i-th node (or edge) in the global best network topology
Objective function	Cost function
Goal	Find the optimal network topology

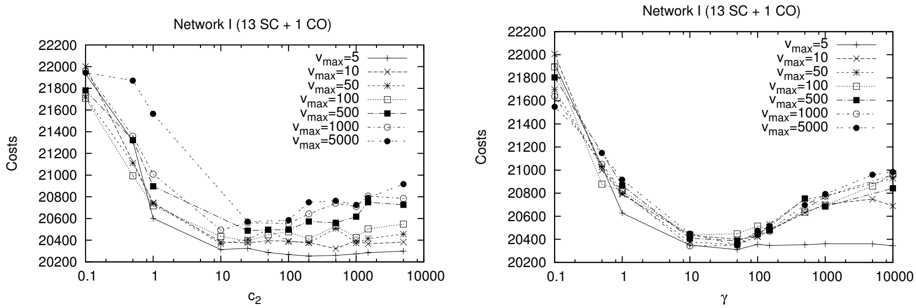
2) Hardware and software: in this approach several heuristic algorithms have been implemented in the standard platform of a planning tool for HFVAN designs in C++. Generally, each parameter or scenario has been tested at least 20 times (runs). *CPLEX* is used to obtain a global or near-global optimum as reference results. The test environment is made up of two parts:

- Heuristic algorithms in C++ are tested at the *HPC* of TU Dresden: *Linux Networx PC-Farm Deimos, CPU: AMD Opteron dual Core 2.6GHz, RAM: 2 GB*;
- *CPLEX* works with a computer: *CPU: Intel(R) Xeon(TM) 3.20GHz, RAM: 4 GB.*

5.2 Results and Comparison

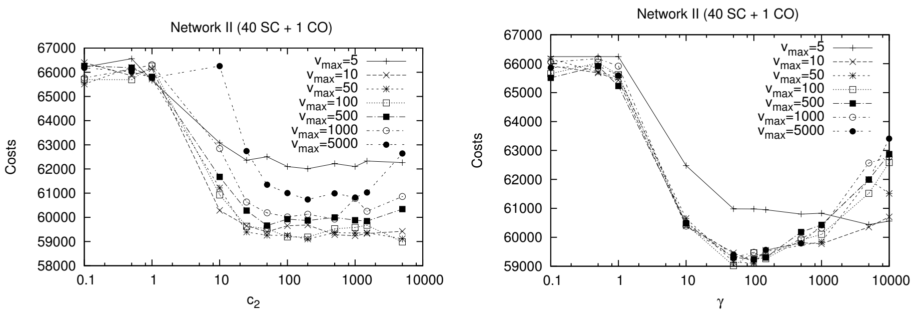
1) Original DBPSO vs. enhanced DBPSO: in the original DBPSO, V_{max} is usually used as a constraint for the update of the velocity. Actually, it plays a relevant role in the optimization process with DBPSO. However, this effect is a negative factor to disturb the performance of DBPSO in HFV access network designs. Fig.5a, Fig.6a and Fig.7a show the relationship between V_{max} and c_2 for Full Particle Modeling (FPM) for Network I/II/III, where c_1=1. Each test point in the figures presents the mean value of optimal costs from at least 20 runs. These results explicitly verify that different V_{max} leads to different results. This effect makes the selection of c_1 and c_2 more complex for the HFV access network topology design. Due to bad solutions, the results of V_{max} ($V_{max} < 5$ for Network I/II, $V_{max} < 10$ for Network III) are not depicted in Fig.5a, Fig.6a and Fig.7a. Hence, they are disregarded for the comparison.

To avoid this negative influence of V_{max}, the enhanced DBPSO is implemented. Fig.5b, Fig.6b and Fig.7b present the optimization results from V_{max}



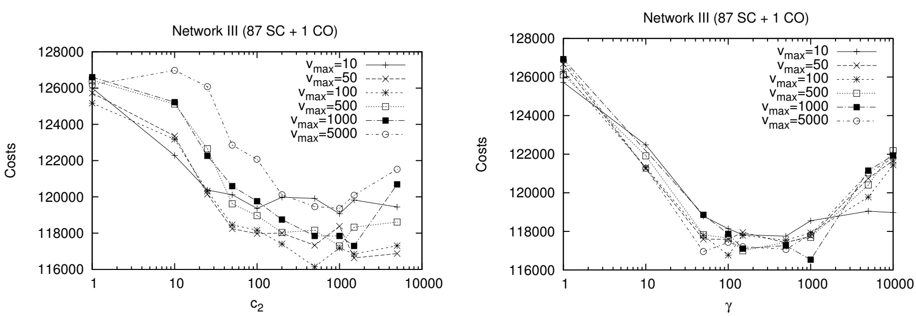
a. DBPSO: V_{max} and c_2 with $c_1 = 1$ b. Enhanced DBPSO: V_{max} and γ

Fig. 5. Comparison of DBPSO and Enhanced DBPSO with the Full Particle Modeling for Network I (5 particles)



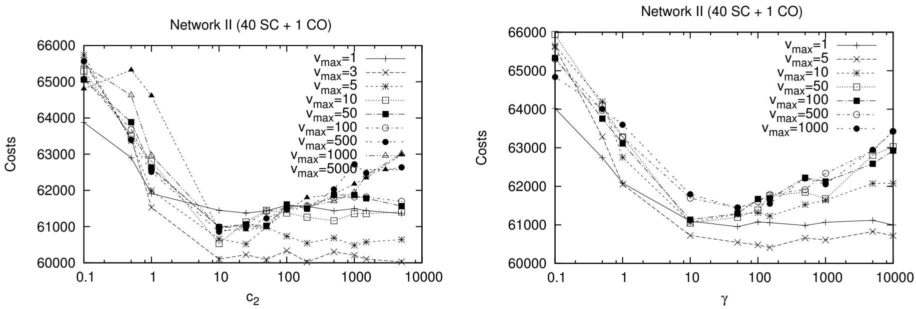
a. DBPSO: V_{max} and c_2 with $c_1 = 1$ b. Enhanced DBPSO: V_{max} and γ

Fig. 6. Comparison of DBPSO and Enhanced DBPSO with the Full Particle Modeling for Network II (10 particles)



a. DBPSO: V_{max} and c_2 with $c_1 = 1$ b. Enhanced DBPSO: V_{max} and γ

Fig. 7. Comparison of DBPSO and Enhanced DBPSO with the Full Particle Modeling for Network III (10 particles)



a. DBPSO: V_{max} and c_2 with $c_1 = 1$ b. Enhanced DBPSO: V_{max} and γ

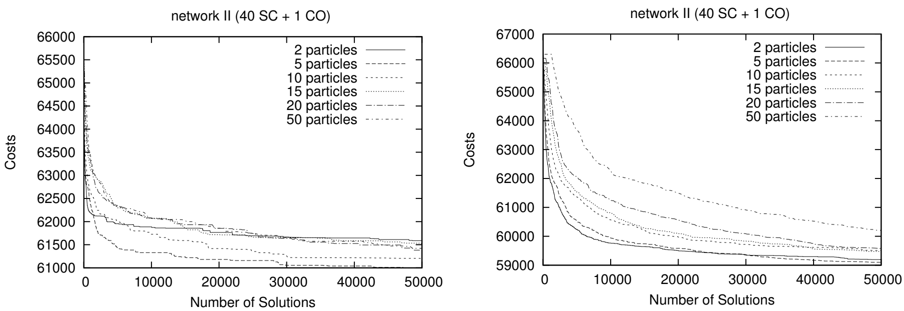
Fig. 8. Comparison of DBPSO and Enhanced DBPSO with the Partial Particle Modeling for Network II (10 particles)

and γ for Network I/II/III with FPM, which are almost independent on V_{max} as γ in $[10, 500]$. Then V_{max} works only as the constraint for the velocity. We do not need to consider it, while studying other parameters, such as the number of particles and iterations. These results identify the advantage of the enhanced DBPSO with γ, which can efficiently work for DBPSO with Full Particle Modeling. A special case appears for Network I and II with V_{max}=5 in Fig.5b and Fig.6b, where V_{max} has no significant influence in these two scenarios with FPM.

The optimization results from DBPSO and enhanced DBPSO with Partial Particle Modeling (PPM) for Network II are depicted in Fig.8. Similar results from PPM for Network I/III are ignored here. However, the improvement of the enhanced DBPSO can not be explicitly seen for Partial Particle Modeling. It stands to reason that only a part of codes are updated by DBPSO with PPM, other binary codes are decided by a few deterministic algorithms. Therefore, the results from DBPSO with PPM for Network I/III are not discussed in this paper.

2) Particle number and visited solutions: if the number of the optimization iterations is fixed, more particles (i.e. visited solutions) at each iteration improve the optimization. Conversely, the optimization time could be increased. Hence, a suitable number of particles is meaningful for the performance of DBPSO. Fig.9 presents the optimization process with different numbers of particles and FPM/PPM for Network II, where the maximal number of solutions is limited. For Network II with FPM or PPM five particles provide the best costs. However, the number of particles depends on the optimization problem. For the HFVAN topology design a higher particle number does not mean that better optimal solutions can be easily found. It stands to reason that more particles could improve the diversification of DBPSO and reduce the intensification. The balance of intensification and diversification of DBSPO has to be obtained subject to empirical results in this work. Hence, the suitable number of particles has to be selected to adapt different real-world problems.

3) Partial Binary Modeling vs. Full Binary Modeling:in terms of Fig.9, DBPSO with FPM has better performance than DBPSO with PPM, because DBPSO with PPM realizing Clustering of LMSs easily leads to trap into local optima. Fig.10



b. Partial Particle Modeling a. Full Particle Modeling

Fig. 9. Comparison of the number of particles with limited solutions ($\gamma = 50$, $V_{max} = 50$)

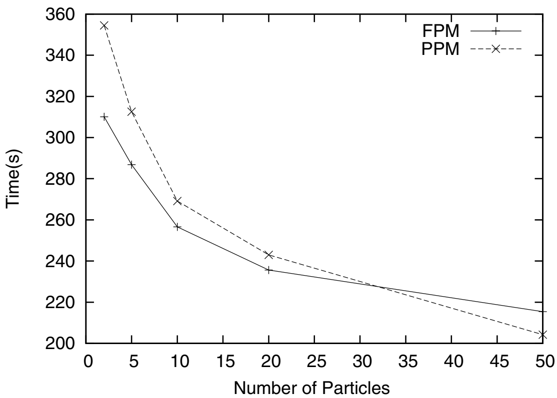


Fig. 10. Comparison of the optimization time for FPM and PPM

compares the optimization time of DBPSO with FPM and PPM. For low numbers of particles the optimization time by DBPSO with FPM is shorter than that by DBPSO with PPM. It stands to reason that a considerable part of the optimization time depends on the repair of the LMSs-BMSs clustering. As mentioned, DBPSO with PPM uses a deterministic method to realize the clustering of LMSs. As assumed, γ is 50, which makes g_{best} more important. During the update of the velocity, other particle are moving to the current best particle. More precisely for HFVAN topology design problems, other network topologies will be changed similar to the best network topology. The best network topology must be a valid solution. Hence, other particles have higher probability to become a valid solution and does not need many reparation for DBPSO with FPM. It can reduce the optimization time. Anyway, the higher number of particles has stronger diversification. By updating the velocity other particles (network topologies) change dynamically, which requires more reparation processes and explicitly increases the optimization time. Therefore, the number of particles and the optimization time should be taken into account simultaneously.

4) Comparison with SA, TS, GA, ACO and PSO with NBM: the number of solutions visited is limited to compare different algorithms. In many papers the comparison of some algorithms is based on the iteration or generation of an optimization process. However, the iteration (i.e.generation) has different meaning or definition for these algorithms. In principle an iteration of trajectory-based search (e.g. SA and TS) means that one valid neighbor or one valid neighboring solution is found. But for population-based search one generation means that the population including a group of valid solutions are changed at the same time. Then the visited solutions by population-based search (e.g. GA, ACO, PSO) at an iteration is much more than those by trajectory-based search at one iteration. The total optimization time depends on the number of solutions and the time needed by each solution. Based on experimental results, the mean time to find one valid solution is similar. Therefore, the limited number of solutions visited is defined to efficiently compare different algorithms in this work.

Tab.2 shows optimization results with maximal solutions 25000/50000/50000 for Network I/II/III. More precisely, SA takes 50/100/100 steps without any change of the temperature as the inner loop and 500 iterations as the outer loop, respectively; TS iteratively finds 5/25/25 neighbors (i.e. solutions) as the

Table 2. Comparison of SA, TS, GA, ACO, PSO with NBM, enhanced DBPSO for HFV Access Network Topology Design

Algorithm	Network	Best	Rel. Dif.	Average	Rel. Dif.	Time(s)
SA	I	20284.0	0.69%	20497.5	1.75%	6.9
	II	59345.1	5.70%	59967.1	6.81%	319.3
	III	113026.0	4.86%	114473.0	6.20%	7963.0
TS	I	20284.0	0.69%	**20284.0**	**0.69%**	3.3
	II	58934.3	4.97%	59229.0	5.49%	335.1
	III	112643.0	4.50%	113634.0	5.426%	9127.2
GA	I	20284.0	0.69%	20375.5	1.15%	2.7
	II	58830.0	4.79%	59492.0	5.97%	119.4
	III	112471.0	4.35%	**113440.0**	**5.25%**	4741.2
ACO	I	20209.9	0.32%	20307.1	0.80%	6.1
	II	58353.9	3.93%	**58918.7**	**4.94%**	192.4
	III	113376	5.18%	114302	6.04%	696.6
PSO	I	20269.3	0.62%	20495.4	1.74%	6.23
	II	58462	4.13%	59777.3	6.48%	331.23
	III	112982	4.82%	115305	6.98%	6728.8
DBPSO	I	**20194.4**	**0.24**%	20372.1	1.13%	7.3
	II	**58031**	**3.36**%	59103.5	5.27%	268
	III	**111426**	**3.38**%	116145	7.75%	5359.3

Candidate List for 2000 steps; GA with Partial Chromosome Modeling generates 50/100/100 new children (i.e. solutions) at each generation, the total generation is 500; ACO fulfills 50/100/100 times Ant-based Clustering to obtain an optimal solution for BMS-LMS layer, where the total generation of Ant-based Positioning is 500 generations; PSO with NBM uses 25/50/50 particles to illustrate solutions for 1000 iterations; DBPSO uses 5/10/10 particles with 5000 iterations to find the required number of solutions.

The relative cost difference to reference optima (Rel.Dif.) can explicitly represent the performance of an algorithm besides the time difference. Reference optima are obtained by *CPLEX*: 20144.1 with 92.96s, 56142.4 with 14938.62s, 107785.8 with 326480.64s for Network I/II/III.

Different optimization results are shown in Tab.2, where the enhanced DBPSO can find the best optima for Network I/II/III and provide excellent average results for Network I/II. The average results of Network III are not satisfying. By reason of the strong diversification and the binary modeling, DBPSO can effectively explore the unvisited space to catch new better solutions. However, it can have negative influence on the search process of DBPSO. For instance, some good solutions have been found, which could be improved by a further search. But the dynamic update of DBPSO could miss these efficient positions. Therefore, the DBPSO can find an excellent solution for the topology design of HFVANs, but the unstable performance could lead the optimization process into local optima. Anyway, the enhanced DBPSO works better than PSO with a non-binary model in Tab.2.

6 Conclusion

This paper presents a strategy for the topology design of hierarchical hybrid fiber-VDSL access networks with heuristic algorithms. Two binary models have been studied for DBPSO. An enhanced update function is proposed to improve the performance and simplify the application of DBPSO. Based on numerical results, the enhanced DBPSO performs better than the original DBPSO in consideration of the maximal velocity. Furthermore, this algorithm can be efficiently used for HFVAN topology design problems, particularly for small or medium-sized networks. It could be helpful for other research in this field.

However, DBPSO strongly depends on the length of particles and initial network topologies. More test networks are required to study them continuously. As a further work, the update function of $x_n(k+1)$ could be studied by another function to avoid the premature convergence. The stability for large scale networks should be studied continuously.

References

1. Zhao, R., Goetze, S., Lehnert, R.: A Visual Planning Tool for Hybrid Fiber-VDSL Access Networks with Heuristic Algorithms. In: The 5th International Workshop on Design of Reliable Communication Networks (DRCN 2005), Island of Ischia (Naples), Italy, pp. 541–548 (2005)

2. Zhao, R., Dai, Q., Lehnert, R.: Planing of Hybrid Fiber-VDSL Access Networks Using Particle Swarm Optimization. In: WTC/ISS (World Telecommunications Congress/International Switching Symposium) and ISSLS (International Symposium on Services and Local Access), Budapest, Hungary (2006)
3. Zhao, R., Liu, H.J., Lehnert, R.: Topology Design of Hierarchical Hybrid Fiber-VDSL Access Networks with ACO. In: The Fourth Advanced International Conference on Telecommunications (AICT2008), Athens, Greece, June 8-13, 2008, pp. 232–237 (2008)
4. Gavish, B.: Topological design of Telecommunication Networks-the Overall Design Problem. European Journal of Operations Research 58(2), 149–172 (1992)
5. Bley, A., Koch, T., Wessaely, R.: Large-scale hierarchical networks: How to compute an optimal architecture? In: Proceeding of Networks 2004, Vienna, Austria, pp. 13–16 (2004)
6. Pioro, M., Medhi, D.: Routing, Flow, and Capacity Design in Communication and Computer Networks. Elsevier Inc., America (2004)
7. Tsenov, A.: Simulated Annealing and Genetic Algorithm in Telecommunications Network Planning. International Journal of Computational Intelligence 2(1), 240–245 (2005)
8. Gavish, B.: Topological design of Telecommunication Networks-local Access Design Methods. Annals of Operations Research 33(1), 17–71 (1991)
9. Godor, I., Magyar, G.: Cost-optimal Topology Planning of Hierarchical Access Network. Computers & Operations Research 32, 59–86 (2005)
10. Chamberland, S., Sanso, B., Marcotte, O.: Topological Design of Two-Level Telecommunication Networks with Modular Switches. Operations Research 48(5), 745–760 (2000)
11. Girard, A., Sanso, B., Dadjo, L.: A Tabu Search Algorithm for Access Network Design. Annals of Operations Research 106(1–4), 229–262 (2001)
12. Eberhart, R.C., Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: Pro. of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, pp. 39–43. IEEE Service Centre, Piscataway (1995)
13. Shi, Y.H., Eberhart, R.C.: A Modified Particle Swarm Optimizer. In: IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, May 4-9 (1998)
14. Eberhart, R.C., Shi, Y.: Particle Swarm Optimization: Developments, Applications and Resources. In: IEEE Proc. of Congress on Evolutionary Computation (CEC 2001), IEEE Service Centre, Seoul Korea (2001)
15. Hu, Y., Eberhart, R.C., Shi, Y.: Engineering Optimization with Particle Swarm. In: Proceeding of IEEE Swarm Intelligence Symposium 2003, pp. 53–57 (2003)
16. Hu, X.H., Shi, Y., Eberhart, R.: Recent Advances in Particle Swarm. In: IEEE Proc. of Congress on Evolutionary Computation (CEC 2004), June 19-23, 2004, pp. 90–97 (2004)
17. Parsopoulos, K.E., Vrahatis, M.N.: Particle Swarm Optimization for Constrained Optimization Problems, Intelligent technologies-theory and applications: new trends in intelligent technologies. Frontiers in Artificial Intelligence and Applications, vol. 76, pp. 214–220. IOS Press, Amsterdam (2002)
18. Sun, S.C., Abraham, A., Zhang, G.Y., Liu, H.B.: A Particle Swarm Optimization Algorithm for Neighbor Selection in Peer-to-Peer Networks. In: CISIM 2007, June 28-30 (2007)
19. Kennedy, J., Eberhart, R.: A Discrete Bindary Version of the Particle Swarm Algorithm. In: Proc. 1997 Conference Systems, Man, Cybernetics, Piscataway, NJ, USA (1997)

20. Chandrasekaran, S., Ponnambalam, S.G., Suresh, R.K., Vijayakumar, N.: A Hybrid Discrete Particle Swarm Optimization Algorithm to Solve Flow Shop Scheduling Problems. In: IEEE Conference on Cybernetics and Intelligent Systems, June 2006, pp. 1–6 (2006)
21. Chen, M., Wang, Z.W.: An Approach for Web Services Composition Based on QoS and Discrete Particle Swarm Optimization. In: Eighth ACIS International Conference on Sotftware Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, July 30 - August 1, 2007, vol. 2, pp. 37–41 (2007)
22. Zhang, C.S., Sun, J.G., Wang, Y., Yang, Q.Y.: An Improved Discrete Particle Swarm Optimization Algorithm for TSP. In: IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology Workshops, No. 5-12, pp. 35–38 (2007)