Availability-Driven Scheduling for Real-Time Directed Acyclic Graph Applications in Optical Grids

Min Zhu, Wei Guo, Shilin Xiao, Anne Wei,, Yaohui Jin, Weisheng Hu, Benoit Geller

Availability-Driven Scheduling for Real-Time DAG Applications in Optical Grids

Min Zhu, Wei Guo, Shilin Xiao, Anne Wei, Yaohui Jin, Weisheng Hu, and Benoit Geller

***Abstract*—Optical grid systems have been viewed as a promising virtual computing environment to support distributed real-time DAG (directed acyclic graph) applications. For such a system involving many heterogeneous computing and network resources, faults seem to be inevitable. Therefore, a fault-tolerant DAG scheduling scheme is necessary to improve the performance of the optical grid system. However, existing joint task scheduling schemes for real-time DAG applications generally do not consider the availability issues when making scheduling decisions. In this paper, we develop an availability-driven scheduling (ADS) scheme that improves the DAG availability iteratively by allocating two copies of one communication task to two disjoint lightpaths for data transfer while satisfying application deadline requirements. Extensive simulation results demonstrate the effectiveness and the feasibility of the proposed scheduling scheme.**

***Index Terms*—Availability, real-time scheduling, distributed applications, fault-tolerant, optical network**

I. Introduction

Recently optical grid systems have emerged as a powerful and cost-effective platform for distributed computing applications such as global e-science, grid computing and collaborative design [1]-[3]. In these applications, the widely distributed and heterogeneous computing resources such as experimental facilities, supercomputers or massive storage systems, are connected by an optical network. An application may be submitted by grid users dynamically via grid middleware. It may have tight real-time requirements that the computation should be finished before a given deadline. Furthermore, these applications usually consist of tens, hundreds, or even thousands of inter-dependent computation tasks and communication tasks, which are modeled by directed acyclic graphs (DAGs). Each computation task can be scheduled to an available grid computing resource for data process. Each communication task can be scheduled to a lightpath for data transfer. Thus joint (computation and communication) task scheduling is an important issue in making cost-effective utilization of optical network and grid resources.

Unfortunately, running real-time DAG applications in such heterogeneous and complex system is susceptible to a wide range of failures as revealed by a recent survey [4]. If an unexpected fault happens, the application will be interrupted and re-executed until the fault disappears. The completion time may then become very large and the application may fail to finish on time. Therefore, employing an effective fault-tolerant scheme provided by the optical grid system is essential to execute reliably real-time DAG applications.

Using a fault-tolerant scheme (e.g., a protection scheme), however, generally consumes more optical network resources for backup lightpaths and induces longer completion time, which might violate the application deadline. The conflicting requirements of good real-time performance imposed by DAG applications and of high Quality of Service (QoS) in a reliable optical network, introduce a new challenge for joint task scheduling schemes.

However, most existing joint task scheduling schemes for DAG applications do not consider the optical network availability issues [5]-[7] and thus they are inadequate for real-time DAG applications. Recently, Liu et al. [8] discussed about failure resilience (protection and restoration) schemes in DAG scheduling problems over optical grid systems. Sun et al. [9] proposed two fault-tolerant policies (overlay and joint approaches) and studied the performance difference of the two policies via simulations. Although the above schemes enhanced the survivability of the optical network integrated computing system by introducing some resource redundancy, they have not investigated the time overhead introduced by these protection schemes and ignore the deadline constraints imposed by the applications. Guo et al. [10] analyzed the fault probability of computing resources and of optical links in a given time duration and proposed a minimal fault probability (MFP) task-scheduling algorithm to minimize the application fault probability. However, the MFP algorithm has not provided any backup resources in practice and real-time DAG applications are interrupted as soon as a fault happens.

This paper considers the problem of fault-tolerant scheduling DAGs with deadline requirements in optical grid systems. This problem is NP-complete because the conventional joint task

This work was supported jointly by the National Science Foundation (NSF) under grant No. 60672016, 60632010, 60572029, and 60825103, and by the 863 Program in China.

Min Zhu, Wei Guo, Shilin Xiao, Yaohui Jin, and Weisheng Hu are with State Key Lab on Fiber-Optic Local Area Networks and Advanced Optical Communication Systems, Shanghai Jiao Tong University, China (e-mail: {zhuminxuan, wguo, slxiao, jinyh, wshu} @ sjtu.edu.cn).

Min Zhu is also with the SATIE lab, Ecole Normale Superieure (E.N.S.) Cachan, 94235, France.

Anne Wei is with the LATTIS Lab, Université de Toulouse II, 31703, Blagnac, France (e-mail: anne.wei@lattis.univ-toulouse.fr).

Benoit Geller is with the UEI Lab, ENSTA ParisTech, 75739, Paris, France (e-mail: benoit.geller@ensta.fr).

scheduling without any fault tolerance is a well-known NP-complete problem [5]-[7]. Thus, we design and evaluate a heuristic approach based on the Availability-Driven Scheduling (ADS) scheme for real-time DAG applications. We focus only on optical link failures since network nodes (e.g., optical cross-connect or amplifier) are usually much more reliable than links [11] [12]. We assume that a *1+1* Dedicated Path Protection (DPP) is available for communication tasks in DAG applications submitted to the optical grid system. For a real-time DAG application, the proposed ADS scheme first verifies whether the application deadline can be met without any availability improvement. If so, the ADS scheme iteratively enhances the application availability level in a cost-effective way under the condition that the availability enhancement does not result in the application deadline violation; otherwise, the DAG application is dropped, because its execution is infeasible. The performance of the ADS scheme is compared with other joint task scheduling schemes via simulations. The simulation results show that the proposed ADS scheme can provide better performance in terms of availability and network resource utilization, while satisfying given real-time requirements.

The rest of this paper is organized as follows; in the next section, we first describe mathematical models for DAG applications and for optical grid systems, and then propose a DAG scheduling extended model to allocate two copies of one communication task to two link-disjoint lightpaths for data transfer. The availability model of DAG applications is also presented. In Section III, we detail the association of the ADS scheduling architecture with the ADS algorithm for real-time DAG applications in optical grids. We display simulation results and evaluate the performance of the ADS scheme in Section IV. Finally, the conclusion is made in Section V.

II. Mathematic Model and Problem Formulation

A. DAG Application Model

Each application is represented by a Directed Acyclic Graph (DAG). A DAG is formulated as $J = (V, E, d)$, where $V = \{v_1, v_2, ..., v_n\}$ represents a set of inter-dependent real-time computation tasks, E is a set of weighted and directed edges used to represent communication tasks among computation tasks, and d is the DAG application's deadline. Each computation task v_i is characterized by a parameter $c(v_i)$, which denotes the amount of data to be processed. The weight on each edge e denotes the volume of data to be transmitted, which is also called as the communication task cost $c(e)$. Note that throughout this paper, the terms application and job are used interchangeably.

Fig. 1 shows an example DAG where each node is assigned an average execution cost, and each edge is assigned a weight. In the DAG, all executed tasks must satisfy task precedence constraints: 1) each *computation task* can be processed only after all its predecessors have finished and all the data needed have been transferred; 2) each *communication task* can start only after that the predecessor computation task is completely finished.

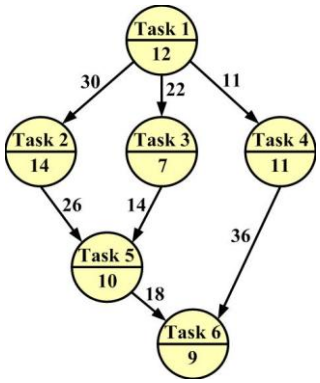

Fig.1 Illustration of a DAG application

B. Optical Grid System Model

The optical grid system shown in Fig. 2 includes some grid computing resources, such as supercomputers, clusters, storages and visualization devices that are used by the DAG application. These grid computing resources are connected by an optical network with Optical Cross-Connect (OXC) nodes and optical links. Each optical link contains several wavelength channels. Each grid resource is attached to one OXC node via a dedicated access link.

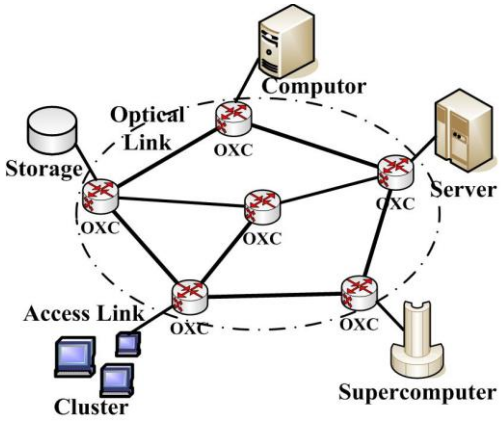

Fig.2 Optical grid system model

We model the optical grid system as a graph $G_{ON} = (S, L, w, G)$, where S is a set of optical switch nodes (i.e. OXC), L is a set of optical links, w is the number of available wavelengths in each optical link, and G is the set of all grid computing resources connected by the optical network. We assume that each optical switch is equipped with all-wavelength converters, and thus there is not any wavelength continuity constraint with the lightpath routing.

C. DAG Scheduling and Constraints

Based on the above models and assumptions, the joint task scheduling algorithm maps each computation task to an available grid computing node for data process and each communication task to one or two lightpaths for data transfer. The DAG scheduling algorithm must satisfy the following constraints:

1) the *grid computing resource constraint*, which ensures

that the computation tasks assigned to the same grid computing node should not be processed at the same time;

2) the *network resource constraint*, which ensures that communication tasks with a same wavelength on a common physical link should not be transferred at the same time.

The objective of the DAG scheduling considered in this paper is to maximize the application availability level without violating the real-time requirement and precedence constraint of the DAG applications. For purpose of simplicity, we only employ the "end" technique [5] in scheduling, which means that the scheduler can only schedule a task starting from time t on the corresponding (network or grid) resources which are free in $[t, \infty]$.

To improve the DAG application availability, a *1+1* Dedicated Path Protection (DPP) scheme is available and some communication tasks may have two copies, called primary copy e^P and backup copy e^B, transferred simultaneously on two link-disjoint lightpaths in optical networks. There are two important parameters to be derived:

1) $t_{ea}(v)$ is the earliest available time for the computation task v; this indicates the time when all data from v's predecessors have arrived;

2) $t_{es}(v)$ is the earliest start time for the computation task v; this additionally signifies that the grid computing resource node $g(v)$ (to which v is allocated) is now available to start the task execution.

Thus, $t_{es}(v) \geq t_{ea}(v)$, and at time $t_{ea}(v)$, the grid computing node $g(v)$ may not be ready to execute task v.

In what follows, we derive the expressions of those time parameters for the DAG scheduling. We first consider a scheduling computation task v_i with one direct predecessor task v_j. Let $f(v_j)$ be the finish time of v_j, $t_{es}(e_{ji})$ the earliest start time of communication task e_{ji}, $c(e_{ji})$ the transmitted data volume, BW the lightpath bandwidth, and $c(e_{ji})/BW$ is the transmission time for communication task e_{ji}. The earliest available time $t_{ea}^{k,v_j}(v_i)$ of the computation task v_i on the k^{th} grid computing resource node of the same type as the computation task is given by the following expression:

$$t_{ea}^{k,v_j}(v_i) = \begin{cases} f(v_j), & \text{if } g(v_i) = g(v_j) \\ t_{es}(e_{ji}) + c(e_{ji})/BW, & \text{otherwise} \end{cases} \quad (1)$$

where $g(v_i)$ is the computing node of task v_i, $t_{es}(e_{ji})$ depends on how the communication task e_{ji} is scheduled on the paths. If the communication task e_{ji} is just assigned to one lightpath without any protection measure, which is referred as *Case1*, the earliest start time of communication task e_{ji} is given by:

$$Case1: t_{es}(e_{ji}) = \max\{t_{ea}(lp(e_{ji})), f(v_j)\} \quad (2)$$

$t_{ea}(lp(e_{ji}))$ is the earliest available time of the lightpath $lp(e_{ji})$ from $g(v_j)$ to $g(v_i)$. However, the communication task e_{ji} can not be executed when a link failure occurs in the lightpath $lp(e_{ji})$.

With the *1+1* DPP scheme, a communication task e_{ji} is provisioned by a primary lightpath $lp(e_{ji}^P)$ and a link-disjoint backup lightpath $lp(e_{ji}^B)$. According to the task precedence constraints, the computation task v_i then must also wait for the data transmission on the backup lightpath to complete. This case is referred as *Case2* and one has the following expression:

$$Case2: t_{es}(e_{ji}) = \max\{t_{ea}(lp(e_{ji}^P)), t_{ea}(lp(e_{ji}^B)), f(v_j)\} \quad (3)$$

In this case, no further operations are required when a link fails (assuming link failures are rare events such that there can be at most one failure during one communication).

We now consider the case of scheduling task v_i with all its direct predecessors. Task v_i must wait until the last data needed from all its predecessors has arrived. Hence, the earliest available time of v_i on the k^{th} grid computing resource node is the maximum of $t_{ea}^{k,v_j}(v_i)$ over all its predecessors:

$$t_{ea}^{k}(v_i) = \max_{e_{ji} \in E}\{t_{ea}^{k,v_j}(v_i)\} \quad (4)$$

With (4), we can obtain the earliest start time $t_{es}^{k}(v_i)$ on the k^{th} grid computing resource node $g_k^r \in G$ of type r by checking if the node is idle since from time $t_{ea}^{k}(v_i)$. The value $t_{es}^{k}(v_i)$ is in turn used to derive $t_{es}(v_i)$, which is the earliest start time for the task on any computing resource node of type r. The expression for $t_{es}(v_i)$ is given as follows:

$$t_{es}(v_i) = \min_{g_k^r \in G}\{t_{es}^{k}(v_i)\}. \quad (5)$$

Consequently, the finish time of the computation task v_i is obtained as follows:

$$f(v_i) = t_{es}(v_i) + c(v_i)/p \quad (6)$$

where $c(v_i)$ is the amount of processed data of the computation task v_i, p is the data processing capability of the grid computing resource node assigned for the computation task v_i. We assume that each grid computing resource node has (the same) one unit data processing capability ($p = 1$).

D. Availability Model and Scheduling Objective

In this paper, we only focus on link failure scenarios and adopt the *1+1* DPP scheme to improve the availability of real-time DAG applications. We assume that different network links fail independently; for any single link, the normal

operating times and repair times are independent processes with known mean values (Mean Time To Failure (MTTF) and Mean Time To Repair (MTTR)). The link availability is calculated as follows [12]:

$$a_j = \frac{MTTF}{MTTF + MTTR} \tag{7}$$

In the case of no path protection (NPP), the overall lightpath availability is the product of all the availabilities of the links along lightpath P. For the *1+1* dedicate path protection (DPP), the availability of a communication task e is computed as:

$$A_e = \begin{cases} \prod_{j \in P} a_j, & \text{NPP} \\ A_p + (1 - A_p) \times A_b, & \text{DPP} \end{cases} \tag{8}$$

where A_p is the primary lightpath availability and A_b is the backup lightpath availability.

Our joint task scheduling algorithm aims at maximizing the entire DAG application availability without violating the DAG's deadline. The proposed ADS scheme has to measure the availability benefit gained by an application. We model the availability benefit as an availability-level function denoted by $AL(DAG): A_e \rightarrow \Re$, where $\Re$ is the summation of the following set of real numbers:

$$AL(DAG) = \sum_{e \in E} w_e A_e \tag{9}$$

In equation (9), w_e is the weight of a communication task $e \in E$ and one has $\sum_{e \in E} w_e = 1$. In this paper, w_e is chosen as the ratio of the communication task cost $c(e)$ of the edge e over the sum of all communication task costs of a DAG application. Thus, we define an optimization formulation to maximize an availability benefit of a real-time DAG application, while assuring that its Scheduling Length SL (also called as completion time) satisfies a deadline d constraint:

$$\text{Maximize } AL(DAG) = \sum_{e \in E} w_e A_e \tag{10}$$

$$\text{Subject to: } SL(DAG) \le d \tag{11}$$

III. Availability-Driven Scheduling Scheme

A. Availability-Driven Scheduling (ADS) Architecture

As depicted in Fig.3, the availability-driven scheduling system architecture consists of an availability-driven joint task scheduler, an execution manager, an admission controller, a resource manager and a fault-tolerant manager. These functions are implemented in a server, which controls in a centralized way each OXC node to setup or release a lightpath for data transfer in the optical network. The server can receive jobs submitted by users via a User Network Interface (UNI).

The admission controller is deployed to perform some feasibility checks by determining if arriving jobs can be completed before their specified deadlines without any availability improvement. An application will be admitted into the system if its deadline can be met; it will be scheduled independently by the ADS algorithm (just after that the last application leaves the system) according to the available grid

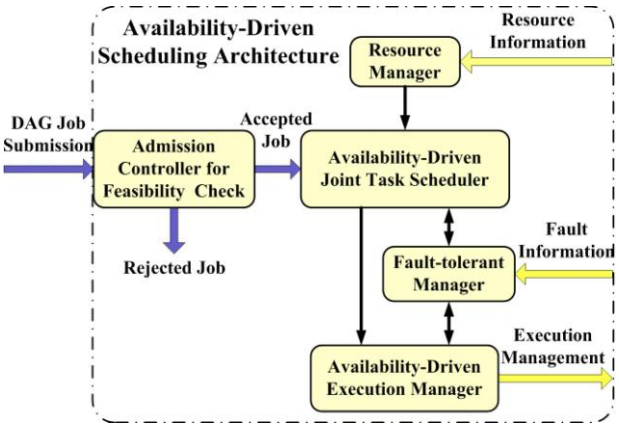

Fig.3 Availability-Driven Scheduling Architecture

computing and network resources. The resource manager maintains and collects the grid computing and network resource information for the joint task scheduler in order to the resource allocation. The availability-driven joint task scheduler makes task scheduling decisions for each application, by maximizing the application availability with the real-time requirements. Based on the task scheduling decisions, the execution manager reserves or releases the corresponding computing and network resources via the control signalling plane. During the system execution, the fault-tolerant manager monitors the optical grid system and adapts certain protection measures to guarantee the jobs to finish on time if some resource failures occur.

B. ADS Algorithm Description

The ADS algorithm outlined in Fig. 4 aims at achieving a higher availability under the two following constraints: 1) increase the availability level without any deadline violation and 2) satisfy the task precedence constraints. In other words, this algorithm tries to iteratively enhance the DAG availability in a cost-effective way under some given real-time requirements.

Initially, the ADS algorithm sorts all the computation tasks in a list according to some priority schemes at step 1. For example, a bottom-level priority is adopted to calculate the bottom level $BL(v_m)$ of each computation task v_m in this paper, which is the length of the longest path from node v_m to a sink task v_s, including the communication cost $c(e_{mn})$. The sink node has no any successor (i.e. $\boldsymbol{SUC}(v_s) = \varnothing$) in the DAG. The set of all direct successors of node v_m is denoted by $\boldsymbol{SUC}(v_m)$. It can be calculated recursively by [13]:

$$BL(v_m) = c(v_m) + \max_{v_n \in \boldsymbol{SUC}(v_m)} \left[BL(v_n) + c(e_{mn}) \right] \tag{12}$$

Before maximizing the availability level of a job, the ADS scheme first verifies whether the feasible schedule is available, just satisfying the deadline and precedence constraints; this can be accomplished by calculating a tentative finish time F_{NPP} without any availability enhancement at step 2. The finish time F_{NPP} is obtained from the conventional DAG scheduling with no path protection (NPP) consideration [5]-[7], which is coherent with *Case 1* of the DAG scheduling model. More

Input: A Real-Time DAG, Deadline = d, an Optical Grid G_{ON}
Output: Schedule Length SL and Availability Level AL
01. Sort the computing tasks $v \in V$ in a list by one sorting policy
02. Calculate a tentative finish time F_{NPP} without any availability improvement considerations
03. **If** $(d \geq F_{NPP})$
04. Calculate a tentative finish time F_{DPP} with each communication task $e \in E$ to be protected
05. **Else Return** $(FAIL)$
06. **If** $(d < F_{DPP})$
07. Initiation finish time $F_{ADS} \leftarrow F_{NPP}$
08. **While** ($F_{ADS} < d$ **and** $\exists e \in E$ not protected) **do**
09. **For** ($e \in E$ **and** e not protected)
10. Compute $\theta_e = \dfrac{w_e * \left(A_e^{DPP} - A_e^{NPP}\right)}{\left(H\left(p_e^{DPP}\right) + H\left(b_e^{DPP}\right) - H\left(p_e^{NPP}\right)\right)}$
11. **End For**
12. Selecte e subject to $\mathrm{Max}(\theta_e)$ and add e to $E^{ADS} \subseteq E$
13. Update the finish time F_{ADS} with set E^{ADS}
14. **End While**
15. Schedule DAG in optical grids with set E^{ADS}
16. **Else** Schedule DAG in optical grids with each communication task $e \in E$ to be protected
17. **Return** $(SL$ and $AL)$

Fig. 4 Availability-Driven Scheduling (ADS) algorithm

specifically, an unscheduled task with the highest bottom level value is selected from the above priority list and is allocated to a type-compatible grid resource node so that it can complete as soon as possible. The optimal resource node on which the task has is allocated with the earliest start time can be found using an exhaustive search among all the resource nodes. Until all tasks are scheduled in optical grid, the schedule length is given out as the result of tentative finish time F_{NPP}.

If the deadline requirement is met (step 3), that is, the finish time F_{NPP} is within the deadline, we further obtain another tentative finish time F_{DPP} with each communication task $e \in E$ to be protected (step 4); otherwise, the DAG job is rejected because of its infeasible schedule (step 5).

It is noted that the tentative finish time F_{DPP} is calculated in a similar way the F_{NPP} is computed. The main difference lies in that with *1+1 DPP* scheme, all communication tasks are protected by a primary lightpath and a link-disjoint backup lightpath. The two primary-backup lightpaths are established simultaneously in optical grid and are all available in the same time duration. Therefore, the grid resource node which has the earliest finish time of data transmission on both primary and backup lightpaths will be chosen to execute the successor computation task. The earliest start time $t_{es}(e_{ji})$ of communication task is expressed just as Eq. (3) in the *Case 2* of the DAG scheduling model.

The relation between the tentative finish time F_{DPP} and the DAG deadline d is the key for invocating or not the iterative optimization process. If the finish time F_{DPP} is also smaller than the deadline d, step 16 executes the DAG scheduling with each communication task $e \in E$ to be protected; otherwise, one will improve the DAG application availability level in an iterative way (from step 7 to 13).

To improve the DAG availability in a cost-effective way, the ADS scheme chooses the most appropriate communication task e to be protected during each *while-iteration* starting at step 8. Specifically, it gives a higher priority to a communication task e with a higher weight that brings a larger availability gain and consumes lower network link resources. Hence, we define the following benefit-cost ratio function θ_e, which measures the increase of availability level with the increase of occupied link resources:

$$\theta_e = \frac{w_e \Delta A_e}{\Delta H(p)} = \frac{w_e * \left(A_e^{DPP} - A_e^{NPP}\right)}{\left(H\left(p_e^{DPP}\right) + H\left(b_e^{DPP}\right) - H\left(p_e^{NPP}\right)\right)} \quad (13)$$

In (13), the numerator represents the weighted enhancement of the availability level, A_e^{DPP} and A_e^{NPP} denote the availability of communication task e in the respective case of *DPP* and *NPP*, whereas the denominator indicates the consumed link resources, $H\left(p_e^{DPP}\right)$ and $H\left(b_e^{DPP}\right)$ being the hop count along the primary path and the backup path for the communication task e in the case of *DPP* scheme.

At step 7, the finish time F_{ADS} is initiated to be F_{NPP}. Under the condition that the finish time F_{ADS} does not violate the DAG deadline, for each unprotected communication task $e \in E$, the ADS scheme identifies the best candidate that has the highest benefit-cost ratio $Max(\theta_e)$, and adds it into the set $E^{ADS} \subseteq E$ (steps 9 to 12). Step 13 updates the finish time F_{ADS} with the set E^{ADS}. Thus, the DAG application availability is enhanced by additionally protecting a communication task e with the highest benefit-cost ratio. The *while-iteration* process continues until that the finish time F_{ADS} is larger than the deadline d or that all communication tasks are protected. In practice, step 15 executes the DAG schedule in optical grids with the final communication-task-protection set E^{ADS}. Finally, we obtain the schedule length SL and the application availability level AL of the submitted DAG job.

C. Time Complexity Considerations

The time complexity for calculating the task bottom level is $O(|V|\log|V| + |E|)$ (step 1) [14] and the time complexity for calculating the tentative finish time at step 2 and step 4 is $O\{|G|(|V| + |E|O(routing))\}$ [13-14], where $O(routing)$ is the time complexity of the routing scheme, V is the set of computation tasks, E is the set of communication tasks, and G

is the set of grid computing resources; $|V|$, $|E|$ and $|G|$ are the number of elements of sets V, E and G respectively. To effectively boost the availability level of an application under the constraints at step 12, the ADS scheme takes an additional $O(|E|)$ time (step 9 to 11) to select the most appropriate communication task as a candidate whose availability will be improved. The time complexities of steps 13, 15 and 16 for joint task scheduling are identical with that of step 3. Thus, the time complexity of the *while-iteration* process is $O\{k|G|(|V|+|E|O(routing))\}$, where k is the number of *while-iteration* processes. Therefore, the time complexity of our ADS scheme is:

$$O(|V|\log|V|+|E|)+O\{k|G|(|V|+|E|O(routing))\} \quad (14)$$

In this paper, the routing scheme is the famous Dijkstra's shortest path algorithm, which is used to calculate the primary and backup lightpaths for communication tasks. Therefore, the time complexity $O(routing)$ is $O(w(|S|\log|S|+|L|))$ [15], where S is a set of optical switch nodes, L is a set of optical links and w is the number of available wavelengths in the optical link. $|S|$ and $|L|$ are the number of elements of the sets S and L, respectively.

IV. Simulation Results and Analysis

To examine the performance of the ADS algorithm, we developed a java-based simulator and compared our scheme with two other heuristic algorithms: ELS_NPP and ELS_DPP. The former uses an Extending List Scheduling (ELS) scheme with No Path Protection (NPP) consideration, and was recently proposed by Wang et al. [5]; ELS is a greedy algorithm used for conventional DAG scheduling without any availability improvement. The latter provides primary and backup paths for each communication task e to improve the DAG application availability. We believe that comparing the ADS scheme with these two joint task scheduling algorithms is meaningful; the availability improvements brought by our scheme will be clearly noticed.

TABLE I Reference System Parameters

Parameters	Reference value
CCR (Communication Computation Ratio)	6,7,8,9
Number of computing tasks in a DAG	100
Mean value of computation task cost	10, (from [6, 14])
Mean number of edges per node	4
MTTR	24h
1/MTTF	300 FIT/km

Each DAG application is represented by a randomly generated DAG. The communication computation ratio (CCR) is defined as the sum of all the communication task costs divided by the sum of all the computation task costs in a DAG. Table 1 summarizes the key system parameters in our simulations. The CCR value in the simulations is chosen as one integer ranging from 6 to 9 and the number of computation tasks is set to 100. The computation task costs per node in a DAG are selected uniformly from 6 to 14 around the mean value 10. The communication task costs are also taken from a uniform distribution, whose mean value depends on the CCR value and on the mean value of computation task cost. The mean number of edge per node is assumed to be 4.

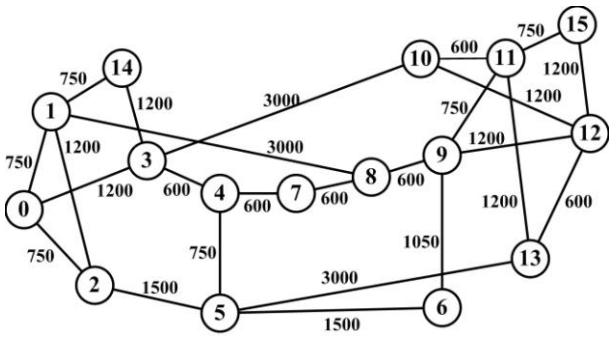

Fig. 5. 16-node NSFNET network topology

The 16-node NSFNET network used in the simulations is shown in Fig. 5. The labels on the links are the lengths of the links in kilometers. Another used network topology is a 16-node ring network and the link length between two adjacent nodes is set to be 100 kilometers. In this paper, the MTTR (Mean Time To Repair) is assumed to be 24 hours and 1/MTTF (Mean Time To Failure) is 300 FIT/km for the above networks (1 FIT = 1 failure in 10^9 hours) [12]. We assume that each optical switch is equipped with all-wavelength converters and has only one grid computing resource connected, which has one unit data processing capability. It is assumed that each optical wavelength channel in each optical link has one unit bandwidth (e.g, 1-Gbps) in the optical grid system and that all the computation tasks and grid computing resources are of three different types. For each case, each scheduling result is the average over 100 simulations.

The performance metrics to evaluate the system performance in our simulations include the following:

1) *Availability*. It is the primary optimal objective for the ADS scheme (see Eq. (10) and (11)).
2) *Communication Task Protection Ratio (CTPR)*. This is the ratio of the number of protected communication tasks $|E^{ADS}|$ ($E^{ADS} \subseteq E$) over the number of all the communication task number $|E|$ in a DAG. Thus, the *CTPR* for an application DAG is calculated as follows:

$$CTPR(DAG) = \frac{|E^{ADS}|}{|E|} \quad (15)$$

3) *Network Resource Utilization (NRU)*. It is defined as the ratio of the total occupied bandwidths over the total supplied bandwidths of all the involved wavelengths and optical link resources during the whole scheduling period.

$$NRU = \frac{\sum_{e_i \in E} \sum_{l \in P(e_i)} BW(\lambda_l(e_i)) \times [ET(e_i) - ST(e_i)]}{\sum_{all\ invovled\ l\ and\ \lambda} BW(\lambda_l) \times SL} \quad (16)$$

In (16), $P(e_i)$ represents one or two (primary and backup) path(s) provisioned for the communication task e_i. $BW(\lambda_l(e_i))$ is the bandwidth of the occupied wavelength λ_l on optical link l for the communication task e_i; the end time and start time of communication task e_i are denoted as $ET(e_i)$ and $ST(e_i)$, respectively. SL is the total scheduling length of the whole DAG application.

4) *Job Complete Time*. This is the job completion time for all the tasks in a DAG. It is also called as scheduling length SL of a DAG application.

A. *Impacts of Different Network Topologies and Available Wavelength Resources*

In the first simulation, we study the performance variations of Job Completion Time, Availability and Network Resource Utilization under six different network scenarios. We choose two different network topologies: the 16-node Ring network and the 16-node NSFNET network with three possible wavelength numbers ($W = 1, 2\ or\ 3$). The CCR is set to be equal to 8. For each network scenario, the deadline value for the ADS scheme is denoted as the character D in Fig. 6(a) and is the average of the job completion time of the ELS_NPP and the ELS_DPP schemes:

$$D = \frac{(F_{NPP} + F_{DPP})}{2} \quad (17)$$

Fig. 6(a) depicts the Job Completion Time of a DAG application versus network topologies under the three different DAG scheduling schemes. The different network topologies and different available wavelength values represent different amount of available network resources. We observe that the DAG completion time becomes shorter as the available wavelength resource increases from 1 to 3 for both the ELS_NPP and the ELS_DPP schemes. That is because more independent tasks in a DAG application can be executed concurrently when more network resources are available. When the available wavelength number is larger than 3, the two scheduling schemes (ELS_NPP and ELS_DPP) have almost the same job completion time. This is due to the fact that more network resources are available for backup lightpaths and the DAG applications' precedence constraints become dominant when the available network resources are sufficient. We also observe that the completion time with the ADS scheme is smaller than the protected ELS_DPP scheme (but larger than the non-protected ELS_NPP scheme), and that it remains within the deadline constraints, leading to a feasible scheme.

Fig. 6(b) only shows the availability performance for the ELS_NPP and ADS schemes. The ELS_DPP scheme usually has higher application availability (10^{-5} unavailability for the ring network and 10^{-3} unavailability for the NSFNET network). It is noted that the ADS scheme significantly enhances the DAG availability at the cost of longer job completion time compared with the ELS_NPP scheme (see Fig.6(a)) and this is especially prominent with the NSFNET network. This is because the ELS_NPP scheme simply shows the baseline performance due to its conservative nature without trying to improve the DAG availability. For the ELS_NPP scheme, we find that a DAG application can obtain a higher availability with the Ring network scenario than with the NSFNET network scenario. This is mainly attributed to the shorter optical link length (100km for each link) and the higher optical link availability of the Ring network. The percentage values indicated with red arrows in Fig. 6(b) are the Communication Task Protection Ratio (CTPR). We find that the larger the CTPR value, the more remarkable the availability performance improvement. The higher CTPR

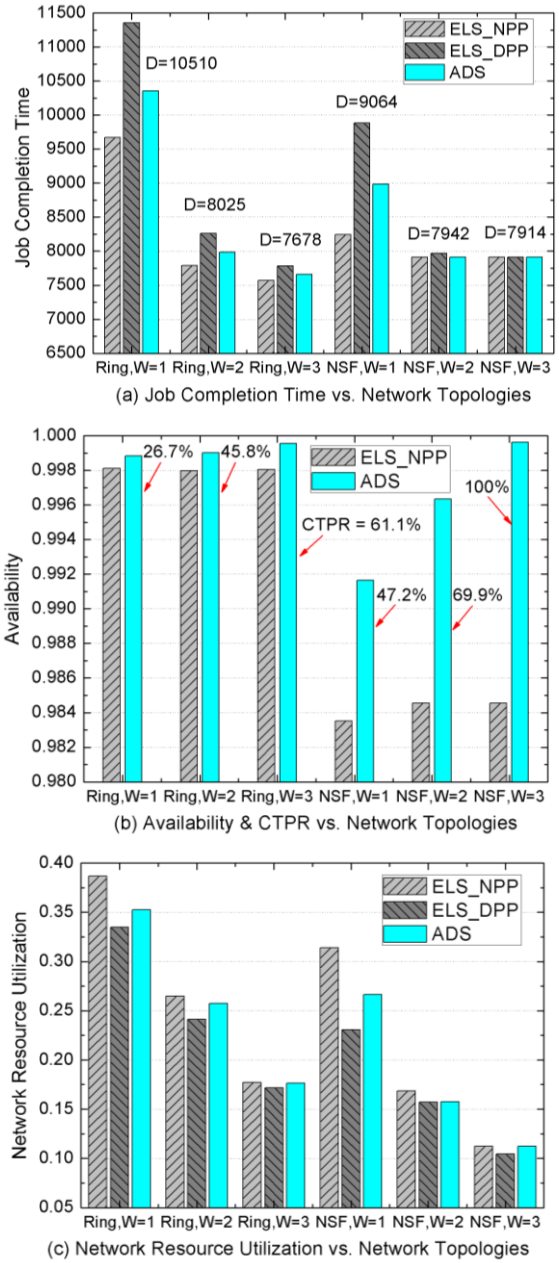

Fig. 6. Performance impacts of different network topologies and available wavelength resources

values usually occur when the ELS_NPP and ELS_DPP schemes have similar completion time. Thus, the smaller additional time relaxation would incur more communication tasks to be protected in DAG scheduling.

From Fig.6(c), we observe that the network resource utilization has the same diminishing trend as the job completion time as the number of available wavelengths per optical link increases. More available network resources, on the one hand, alleviate the network link resource contentions and shorten the completion time; on the other hand, the inherent precedence constraints of the DAG application become increasingly a dominant feature, which keeps more involved network resources idle during most of the execution period of a DAG application. For each network scenario, the ELS_NPP scheme achieves the highest network resource utilization and the ELS_DPP scheme has the lowest utilization. This also indicates that more protected communication tasks implies lower network resource utilization. The reason is that ELS_DPP scheme consumes double network resources for each communication task compared to the ELS_NPP scheme, which in turn reduces the number of independent tasks to be executed concurrently and results in more spare time of involved network resources.

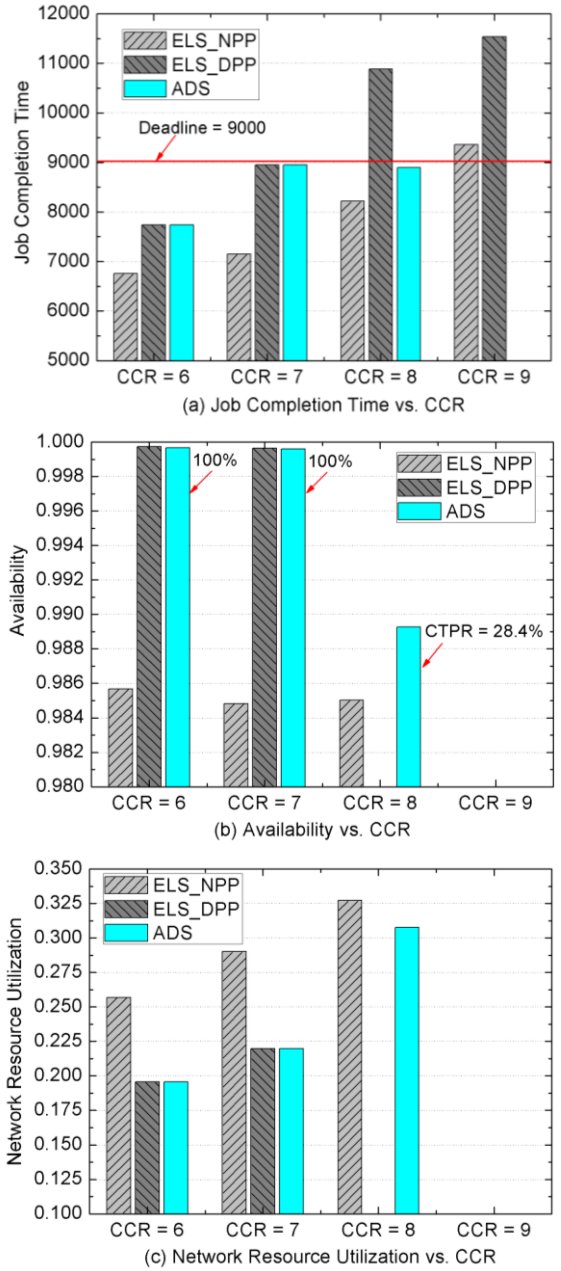

Fig. 7. Performance impact of the CCR on (a) Job Completion Time, (b) Availability and CTPR, (c) Network Resource Utilization

B. *Impact of the Communication-Computation Ratio (CCR)*

We now study the impact of the CCR variations on the DAG schedule results and performances for the three different scheduling schemes. The simulations are carried out over the NSFNET network with one available wavelength, but similar conclusions can be made with other network topologies. The DAG application deadline is set to 9000 as shown in Fig. 7(a).

As the CCR is increasing form 6 to 9 in Fig. 7(a), the job completion time for both the ELS_NPP and ELS_DPP schemes becomes larger, because the communication task sizes of a DAG application are also increasing. The ADS scheme always generates feasible scheduling results within the deadline as long as the ELS_NPP scheme is also feasible. When the CCR is equal to 9, the job completion time of both the ELS_NPP and the ELS_DPP schemes exceed the deadline. Hence the ADS scheme could not generate a feasible schedule and the corresponding position is left blank. Similarly, when any of the three scheduling schemes was not able to generate feasible scheduling results satisfying the DAG deadline requirement, the space is left blank in Fig. 7(b) and Fig. 7 (c).

In Fig.7(b), the percentage values indicate the CTPR. We observe that the ADS scheme obtains the best availability performance, and even a 100% CTPR, like the ELS_DPP scheme, in the case of a CCR equal to 6 or 7. This is because the deadline constraint is relatively loose and there is enough time used to protect all communication tasks in a DAG. When the CCR is set to 8, the ADS scheme partly protects 28.4% of the communication tasks in a DAG and enhances the availability under the deadline constraints, contrarily to the ELS_NPP scheme.

Fig.7(c) plots the network resource utilization of the different scheduling schemes for various CCR values. We can see that the ELS_NPP scheme has the best performance among the three scheduling schemes in each situation. The reason for this is exactly the same as already observed in Fig.6 (c). We can also notice that the network resource utilization of each scheme increases as the CCR value increases. This can be explained as follows: as the load of communication task is heavier when the CCR value increases, consequently, the occupied duration of network (link) resources and the whole DAG's scheduling length (job completion time) are similarly increased by the same amount of time. According to the definition of the network resource utilization (see Eq. (16)), the utilization ratio becomes higher and higher as the CCR increases.

C. *Impact of the Application Deadline*

Finally, we study the impact of the deadline on the

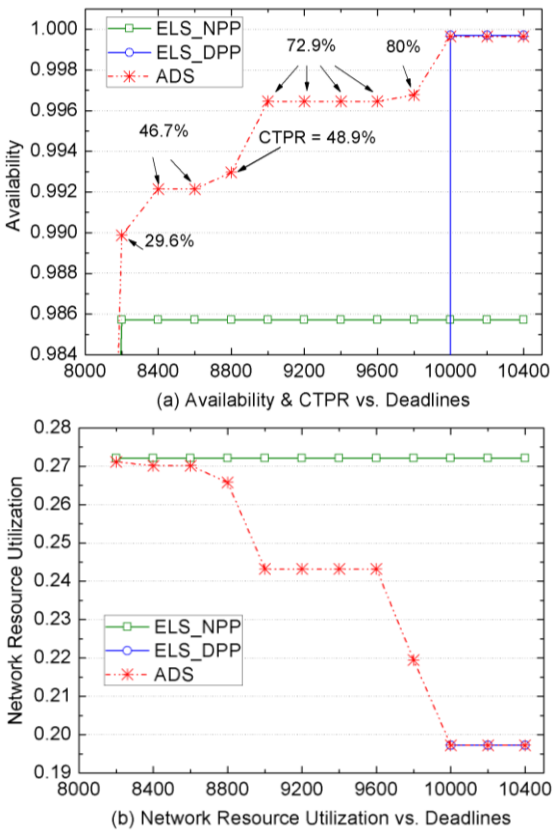

Fig. 8. Performance impact of the deadline on (a) Availability and CTPR, (b) Network Resource Utilization

availability and network resource utilization for the three different scheduling schemes. The simulations are carried out over the NSFNET network with one available wavelength. The CCR is equal to 8. In this context, the value of the deadline is between that of the job completion times of the ELS_NPP scheme and of the ELS_DPP scheme, and this facilitates the study the deadline variation impact on the ADS scheme. For a tested DAG application with 100 computation tasks, the job complete time is respectively equal to 8228 for the ELS_NPP scheme, and equal to 10023 for the ELS_DPP scheme. The DAG application deadline for the ADS scheme varies from 8000 to 10400.

The ADS scheme fully exhibits its better performance over the ELS_NPP scheme, and the ELS_DPP scheme is not even able to produce feasible schedule results when the application has a relatively tight deadline in Fig. 8(a). As the deadline becomes looser, the availability and communication task protection ratio (CTPR) with the ADS scheme is rising. This is because more communication tasks in a DAG can be protected within the looser deadline. When the deadline is larger than 10023, the ELS_DPP scheme can produce feasible schedule results, and even then the ADS scheme achieves, like the ELS_DPP scheme, the same best availability performance.

In Fig. 8(b), when the deadline is about 8200, both the ADS scheme and the ELS_NPP scheme have high network resource utilization (about 27.2%). As the deadline becomes looser, the network resource utilization of the ADS scheme is lower than that of the ELS_NPP scheme and becomes smaller and smaller. When the deadline is larger than 10023 (the completion time of the ELS_DPP scheme), the ADS scheme still achieves with the ELS_DPP scheme the lowest network resource utilization (about 19.8%). This means that the communication task protection consumes more network resources, which in turn reduces the number of independent tasks to be executed concurrently, thus resulting in lower network resource utilization.

These simulation results clearly demonstrate that the optical grid system can even gain more performance benefits with our ADS scheme when the real-time DAG applications have relatively tight deadline requirements. This is because the ADS scheme is self-adaptive compared to the ELS_NPP and ELS_DPP schemes which have no flexibility on the job deadline requirements.

V. Conclusions

In this paper, we present a fault-tolerant joint task scheduling scheme for real-time DAG applications over the optical grid systems considering optical link failure scenarios. The mathematical models describing the DAG applications and the optical grid system are given. An extended joint task scheduling model taking into account the communication task protection is developed. Furthermore, we propose an Availability-Driven Scheduling (ADS) scheme that improves the availability iteratively under the application deadline requirements by allocating two copies of one communication task to two link-disjoint lightpaths for data transfer. The performance of our ADS scheme is evaluated for different network scenarios and is compared with two other joint task scheduling schemes: ELS_NPP and ELS_DPP.

The simulation results show that the ADS algorithm can provide better performances for real-time DAG applications when optical link failures occur. The ADS scheme exhibits larger job availability than the ELS_NPP scheme at the small price of a larger completion time which becomes neglectable when the network resources (number of wavelengths) are sufficient. The ELS_DPP scheme has even a better availability than the ADS scheme, but it can not be used in practice because it is not able to support the deadline constraints, contrarily to the self-adaptive ADS scheme. We thus believe that the ADS scheme is a good candidate to provide reliable real-time DAG applications over optical grid systems.

Acknowledgment

The authors want to thank the anonymous reviewers for their constructive comments to improve the quality of our paper.

References

[1] D. Simeonidou, R. Nejabati, G. Zervas, D. Klonidis, A. Tzanakaki, and M. J. O'Mahony, "Dynamic Optical-Network Architectures and Technologies for Existing and Emerging Grid Services," J. Lightwave Technol. vol. 23, pp. 3347–3357, 2005.

[2] W. Guo, Y. Jin, W. Sun, W. Hu, X. Lin, M. Wu, H. Liu, S. Fu and J. Yuan, "Distributed Computing over Optical Networks," Optical Fiber Communications (OFC), San Diego, California, USA, Feb. 2008.

[3] T. Lehman, J. Sobieski, B. Jabbari, “DRAGON: A Framework for Service Provisioning in Heterogeneous Grid Networks,” IEEE Commun. Mag,, vol. 44, no. 3, pp. 84-90, March 2006.
[4] R. Medeiros, W. Cirne, F. Brasileiro, J. Sauve, “Faults in Grids: Why are they so bad and What can be done about it?,” in the 4th Workshop on Grid Computing, pp. 18-24, Tokyo, Japan, May 2003.
[5] Y. Wang, Y. H. Jin, W. Guo, W. Q. Sun, W. S. Hu, and M. Y. Wu, “Joint Scheduling for Optical Grid Applications,” J. Opt. Netw. Vol. 6, pp. 304–318, 2007.
[6] Z. Sun, W. Guo, Z. Wang, Y. Jin, W. Sun, W. Hu, and C. Qiao, “Scheduling Algorithm for Workflow-based Applications in Optical Grid,” J. Lightwave Technol. vol. 26, pp. 3011-3020, 2008.
[7] X. Liu, W. Wei, X. Yu, C. Qiao, T. Wang, “Distributed Computing Task Assignment and Lightpath Establishment (TALE),” IEEE High Speed Networks (HSN) workshop, collocated with IEEE INFOCOM 2007, Anchorage.
[8] X. Liu, C. Qiao, T. Wang, “Survivable Optical Grids,” Optical Fiber Communication (OFC) Conference, San Diego, California, USA, paper OWN1, Feb. 2008.
[9] Z. Sun, W. Guo, Y. Jin, W. Sun, W. Hu, "Fault-tolerant Policy for Optical Network Based Distributed Computing System," IEEE International Symposium on Cluster Computing and the Grid (CCGrid), Lyon, France, May, 2008.
[10] W. Guo, Z. Liang, Z. Sun, S. Xiao, Y. Jin, W. Sun, and W. Hu, "Task Scheduling Considering Fault Probability for Distributed Computing Applications over an Optical Network," J. Opt. Netw. Vol. 7, pp. 947-957, 2008.
[11] L. Song and B. Mukherjee, “Impacts of Multiple Backups and Multi-Link Sharing among Primary and Backups for Dynamic Service Provisioning in Survivable Mesh Networks,” Optical Fiber Communications (OFC), Anaheim, CA, March 2007.
[12] J. Zhang and B. Mukherjee, “A Review of Fault Management in WDM Mesh Networks: Basic Concepts and Research Challenges,” IEEE Networking. Vol. 18, no. 2, pp. 41–48, 2004.
[13] O. Sinnen and L. Sousa, “Communication contention in task scheduling,” IEEE Trans. Parallel Distrib. Syst.. Vol.16, pp. 503–515, 2005.
[14] O. Sinnen and L. Sousa, “List scheduling: extension for contention awareness and evaluation of node priorities for heterogeneous cluster architectures,” Parallel Comput. Vol. 30, pp. 81–101, 2004.
[15] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, Introduction to Algorithms, MIT, 1990.

Min Zhu received the Master of Science degree in optical telecommunications engineering from Shanghai University in 2007. He is currently pursuing a joint PhD degree both in Shanghai Jiao Tong University and in Ecole Normale Supérieure de Cachan (ENS de Cachan), France.
His research interests include optical access network and systems, optical grid, network resource scheduling and optimization algorithm. He also serves as a reviewer for IEEE Journal of Lightwave Technology.

Wei Guo is an associate professor of state key lab on fiber-optic local area networks and advanced optical communication systems in Shanghai Jiao Tong University since 2003. Before she entered in SJTU, she was a senior engineer and a project manager of the Fiber home Telecommunication Technologies CO., LTD from 2001-2003. She has over 50 publications published in technical journals and conferences. Her research interests include Optical Grid, network planning, and optimization algorithm.

Shilin Xiao received the M.S. degree from the University of Electronic Science and Technology of China, Shanghai, and the Ph.D. degree from Shanghai Jiao Tong University, in 1988 and 2003, respectively.
He is currently a Professor with state key lab on fiber-optic local area networks and advanced optical communication systems in Shanghai Jiao Tong University. His research interests are in the area of all optical communications, especially optical amplifier and switching.

Anne Wei graduated from the Department of Electronic Engineering of the Shanghai University in 1986 and completed her PhD in 1999 at Institute National des Telecommunications, France. After two years spent working for STERIA a computing system company and eight years as an Assistant professor at the University of Paris XII, she is now a Professor at the University of Toulouse II where she teaches computing science and networks. Her research interests include computer networks, post-3G mobile systems and especially security, QoS and mobility in wireless network communications.

Yaohui Jin is a professor in the state key lab on fiber-optic local area networks and advanced optical communication systems, Shanghai Jiao Tong University, China. Prior to joining SJTU, he was a member of technical staff at Bell Labs Research China from 2000 to 2002. He served as the TPC member in many international conferences. He published more than 50 papers in technical journals and conferences. His research interests include optical networking, Optical Grid and switch scheduling.

Weisheng Hu is the professor and director of the state key lab on fiber-optic local area networks and advanced optical communication systems, Shanghai Jiao Tong University. His interests are on generalized automatic switched optical network, and optical packet switching. He is the author or co-author of over 100 journal and conference papers.

Benoît Geller received the M.Sc. degree in electrical engineering from the ENSEIRB in 1988, the Ph.D. in telecommunications from the INPG in 1992, and as an Associate Professor at University Paris 12, the Accreditation to Supervise Research Habilitation à Diriger des Recherches in information sciences in 2004.
He is presently head of the Multisensor and Information Team at SATIE laboratory (Ecole Normale Superieure de Cachan, Chachan, France). He is currently working on iterative methods with application to telecommunications. He also has a strong experience in digital communications systems with emphasis on mobile and broadband systems (OFDM, CDMA, etc.), channel coding (turbo codes, algebraic codes), and signal processing applied to telecommunications (synchronization, equalization, etc.). He has been involved in several European projects and has also been in charge of many industrial projects. He has published about 30 international papers and has four patents pending.