Agència de Gestió d'Ajuts Universitaris i de Recerca

Memòria justificativa de recerca de les convocatòries BE, PIV, BCC, NANOS i BP

La memòria justificativa consta de les dues parts que venen a continuació:
1.- Dades bàsiques i resums
2.- Memòria del treball (informe científic)

Tots els camps són obligatoris

1.- Dades bàsiques i resums

Nom de la convocatòria

BE

Llegenda per a les convocatòries:

BCC	Convocatòria de beques per a joves membres de comunitats catalanes a l'exterior (BCC)
BE	Beques per a estades per a la recerca fora de Catalunya (BE)
BP	Convocatòria d'ajuts postdoctorals dins del programa Beatriu de Pinós (BP)
NANOS	Beques de recerca per a la formació en el camp de les nanotecnologies (NANOS)
PIV	Beques de recerca per a professors i investigadors visitants a Catalunya (PIV)

Títol del projecte: ha de sintetitzar la temàtica científica del vostre document.
Label Space Reduction in All-Optical Label Switchiing Networks

Dades de l'investigador

Nom	Cognoms
Fernando	**Solano Donado**

Correu electrònic
fsolanod@eia.udg.es

Dades del centre d'origen

Comunicaciones y Sistemas Distribuidos
Departamento de Electrónica, Informática y Automática
Universidad de Girona

Número d'expedient
2006 BE 00272

Paraules clau: cal que esmenteu cinc conceptes que defineixin el contingut de la vostra memòria.
Computer Networks, All-Optical Packet Switching Networks, GMPLS, Label Space Reduction

Data de presentació de la justificació
2007-09-04

Generalitat de Catalunya
Departament d'Innovació, Universitats i Empresa

Resum del projecte: cal adjuntar dos resums del document, l'un en anglès i l'altre en la llengua del document, on s'esmenti la durada de l'acció

Resum en la llengua del projecte (màxim 300 paraules)
All-Optical Label Swapping (AOLS) forms a key technology towards the implementation of All-Optical Packet Switching nodes (AOPS) for the future optical Internet. The capital expenditures of the deployment of AOLS increases with the size of the label spaces (i.e. the number of used labels), since a special optical device is needed for each recognized label on every node. Label space sizes are affected by the wayin which demands are routed. For instance, while shortest-path routing leads to the usage of fewer labels but high link utilization, minimum interference routing leads to the opposite.

This project studies and proposes All-Optical Label Stacking (AOLStack), which is an extension of the AOLS architecture. AOLStack aims at reducing label spaces while easing the compromise with link utilization. In this project, an Integer Lineal Program is proposed with the objective of analyzing the softening of the aforementioned trade-off due to AOLStack. Furthermore, a heuristic aiming at finding good solutions in polynomial-time is proposed as well. Simulation results show that AOLStack either a) reduces the label spaces with a low increase in the link utilization or, similarly, b) uses better the residual bandwidth to decrease the number of labels even more.

Resum en anglès(màxim 300 paraules)

All-Optical Label Swapping (AOLS) forms a key technology towards the implementation of All-Optical Packet Switching nodes (AOPS) for the future optical Internet. The capital expenditures of the deployment of AOLS increases with the size of the label spaces (i.e. the number of used labels), since a special optical device is needed for each recognized label on every node. Label space sizes are affected by the wayin which demands are routed. For instance, while shortest-path routing leads to the usage of fewer labels but high link utilization, minimum interference routing leads to the opposite.

This project studies and proposes All-Optical Label Stacking (AOLStack), which is an extension of the AOLS architecture. AOLStack aims at reducing label spaces while easing the compromise with link utilization. In this project, an Integer Lineal Program is proposed with the objective of analyzing the softening of the aforementioned trade-off due to AOLStack. Furthermore, a heuristic aiming at finding good solutions in polynomial-time is proposed as well. Simulation results show that AOLStack either a) reduces the label spaces with a low increase in the link utilization or, similarly, b) uses better the residual bandwidth to decrease the number of labels even more.

2.- Memòria del treball (informe científic sense limitació de paraules). Pot incloure altres fitxers de qualsevol mena, no més grans de 10 MB cadascun d'ells.

1 Background in Brief: The LASAGNE Project

The LASAGNE project was originally conceived for Optical Burst Switching (OBS) networks; however its success became for its adaptation to Optical Packet Switching (OPS) networks, which clearly need more label processing that the former technology.

The LASAGNE project has as purpose the design of a device that is in charge of all-optical label swapping, named an All-Optical Label Swapping (AOLS)-block (see Fig. 1). An AOLS-block is needed for each wavelength port in the OPS node that needs packet routing. An AOLS-block is in charge of reading the incoming label of a packet, replace it by the proper outgoing label (label swapping) and convert the packet to its respective outgoing wavelength color. It is worth noting that all these operations are performed optically in the LASAGNE project.

Optical Correlator Block. When a packet enters the AOLS-module, its payload is separated at 40Gbs [BPY$^+$02]. What remains of the packet, i.e. the optical label, is fed to the *optical correlator* block. Within the *optical correlator* block, the incoming label is replicated several times, so every All-Optical Logic XOR Gate (AOLXG) [MRM$^+$02] has a duplicate of the incoming label. An AOLXG is an optical device, implemented with a Semiconductor Optical Amplifier-based Mach-Zender Interferometer (SOA-MZI) that compares two labels and emits a high-intensity light pulse upon matching.

Local Address Generation Block. In order to perform the matching of the incoming label with the different recognizable labels of the switch, all these recognizably local labels are generated in parallel by the *local address generation* block - at the same time the incoming label is replicated. Each of these local labels is given to a different AOLXG. As a consequence, each AOLXG matches, in parallel, the same incoming label of the packet with a different locally "stored" label. Upon matching, the proper correlator transmits a high intensity light pulse. So far, the optical label has been identified. The high-intensity light pulse is sent to both the *new label generation* block and to the *control* block.

New Label Generation Block. The *new label generation* block produces the corresponding output label of the packet. This new label is inserted behind the payload.

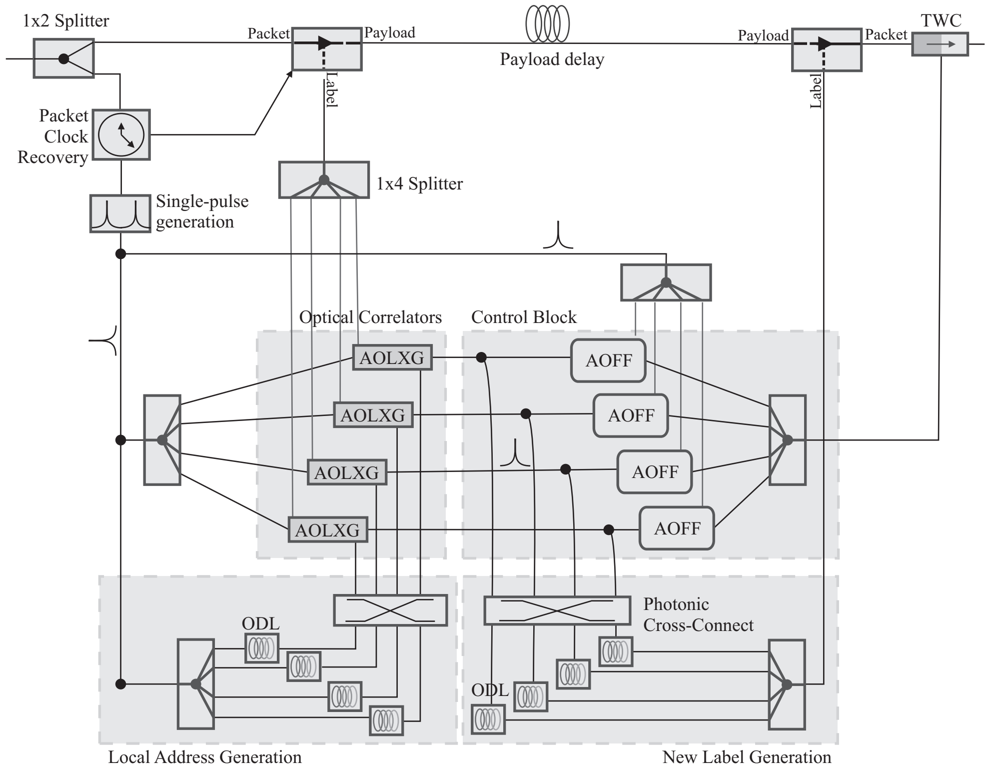


Figure 1: AOLS block able for label swapping

Within the LASAGNE AOLS-block architecture, two label-generation blocks are needed: the *local address generation* block and the *new label generation* block. As mentioned before, the first one is in charge of generating the set of recognizable incoming labels, which feed later the correlators. The second one is in charge of generating the set of outgoing labels. To generate a label (either incoming or outgoing), an Optical Delay Lines (ODL) is used. An ODL is a very simple device comprised of several Fiber Delay Lines (FDL), couplers and splitters.[1]

Control Block. The *control* block drives the wavelength converter, so the output packet (including the new label) can be tuned to its proper outgoing wavelength frequency. As a consequence, both the payload and the header are wavelength converted. The *control* block is implemented with All-Optical Flip-Flops (AOFF). Depending on the matching address (indicated by one of the pulses coming from the correlators), the appropriate flip-flop will emit a Continous Wave (CW) signal at a certain wavelength. The frequency emitted by each AOFF is fixed. After the new label has been inserted, the packet is converted to the wavelength generated by the flip-flop.

Finally, the packet is routed by means of an Arrayed Waveguide Grating (AWG). Therefore, the wavelength on which the packet leaves the AOLS-block determines the outgoing port of the node.

Two Photonic Cross-Connect (PXC) are used inside the AOLS-block to provide label swapping (inside the *new label generation* block) and wavelength conversion (inside the *local address generation* block) flexibility. The first PXC - in the *new label generation* block - switch the matching pulse from any AOLXG to any ODL. This offers a switching matrix between incoming (matching pulses) and outgoing labels (ODLs). The second PXC - in the *local address generation* block - switch the generated addresses between AOLXG correlators. This switching affects (by selecting) the fiber line through which the matching pulse is propagated. As a consequence, since each AOFF generates a CW fixed wavelength, the PXC selects the CW frequency generated by the AOFF fed by the matching pulse. These two PXC are part of the network control plane and are low-speed dynamically configurable.

1.1 Label Spaces and Contention Resolution

In AOLS label spaces can be defined using one of the following policies.

- *Global.* There is a single space of labels for all the traffic in the switch. Forwarding decisions are taken solely using this label. Since there is only one label space per switch in this case, the number of labels needed is bigger than with any of the other options.

[1]Upon the arrival of one pulse, the pulse is split in the number of ones that are needed for the generated label. For instance, to generate the number five (101 in binary), the pulse is split in two pulses (one for each one in the binary string). Then, different delays (using FDLs) are given to each one of the split pulses, so the third pulse has a greater delay than the second one, and so on. Finally, all the split pulse are coupled again in one light stream.

- *Per Fiber.* Separate label spaces are given for each fiber. Forwarding decisions are taken using the incoming fiber port and the label.
- *Per Wavelength Color.* Separate label spaces are given for each wavelength color. Forwarding decisions are taken using the incoming fiber port and the label.
- *Per Wavelength.* Different label space are used per each wavelength, i.e. wavelength color + fiber port. Therefore, forwarding decisions are taken considering the tuple wavelength color, incoming fiber port and label. It provides the smaller label spaces.

Although a *per wavelength* label space lead to the least number of AOLS blocks, it decreases the most the OPS performance when contention resolution is needed. Packet contentions are usually present at the output of the switch. Contention occurs in an OPS when two two packets are competing for the same output wavelength in the same output fibre port at the same time. Since all wavelengths belonging to a fiber lead to the same node, the most used method for contention resolution in OPS is to use a different wavelength in the same fiber to forward the packet in mention. However, in this case, a *per wavelength* label space would lead to an incorrect forwarding since packets may arrive on any wavelength (as a result of a previous contention resolution). The same would happen if a *per wavelength color* label space is used. In this contention resolution scheme is more appropriate to use a *global* or *per fiber* label space.

The use of wavelengths for contention resolution implies the use of several wavelength tunable converters. To avoid the use of wavelength tunable converters, a different approach is often considered for contention resolution. When two packets are competing for the same wavelength in the same fiber, instead of dropping one of the packet, the packet is sent using another fiber port; a principle known as *deflecting routing* or *hot-potato routing*. Neighboring nodes must be capable of handling the packet in question in order to reach its original destination. In this contention resolution scheme a *per wavelength color* or ever a *per wavelength* label spaces can be used.

Since an analysis of different contention resolution schemes in OPS is out of the scope of this dissertation, henceforth, as a worse case scenario, label spaces on a *global* basis are the subject of this dissertation.

It should be pointed out that the number of AOLS blocks is independent of the policy to define the label space. While the number of labels depends on how many demands are switched in the OPS, the number of AOLS blocks depends on how many wavelengths and fiber ports are used by the OPS. However, both of them define the cost and the flexibility of the architecture.

1.2 Label Stripping

Label stripping can be seen as a special case of label stacking, discussed previously. In label stripping, packets header are comprised of a stack full of labels. At each hop, the node strips off one label, i.e. pop the stack, and process it.

The content of the stripped label is used for forwarding, as usual, but labels are never swapped.

Since each node pops the stack once, there is one different label in the stack per each hop. In order to improve resource utilization, a label is used to indicate the link that must be taken to forward the packet. Therefore, each node must handle a minimum number of labels equal to the number of neighbors it has. The number of AOLXG is drastically reduced.

On the other hand, since the stack size should be equal to the number of hops a packet would traverse, the size of the stack is augmented. This would incur in a waste of bandwidth, since the allocated space for the stack cannot be recovered even though the stack contains only one optical label.

1.3 Performance Overview

In this subsection a subset of the results presented by Van Caenegem *et al.* in [CCPD06] are presented. The simulations were performed using the European Network shown in Fig. 3 and the network demands of [MCL+02].

Initially, the dimension of the AOLS-block is studied considering the different types of labels spaces previously mentioned, these are: *global*, *per wavelength*, *per wavelength color* and *per fiber*. Sizing a AOLS-block implies counting:

- the number of PXC ports used in the *new label generation* module,
- the number of PXC ports used in the *local generation address* module,
- the length (distance) of all the incoming ODL,
- the length (distance) of all the outgoing ODL,
- the number of AOLXG correlators used,
- the number of outgoing ODL,
- the number of bits used to encode labels

The results are grouped together differently in Fig. 4 and Fig. 5, with the purpose of depicting different trade-offs.

1.3.1 The Cost of Contention Resolution using Wavelengths

Fig. 4 shows the reduction percentage of the number of used optical components when the label spaces are unable to support contention resolution.

For instance, the first bar (darker color) in Fig. 4 represents the reduction ratio of the optical components when the *per wavelength color* label space is used and the *global* label space, which is its contention-resolution counterpart. In the same figure, the second bar (lighter color) resumes the optical components reduction ratio when a *per wavelength* label space and, its counterpart, *per fiber* label space are used.

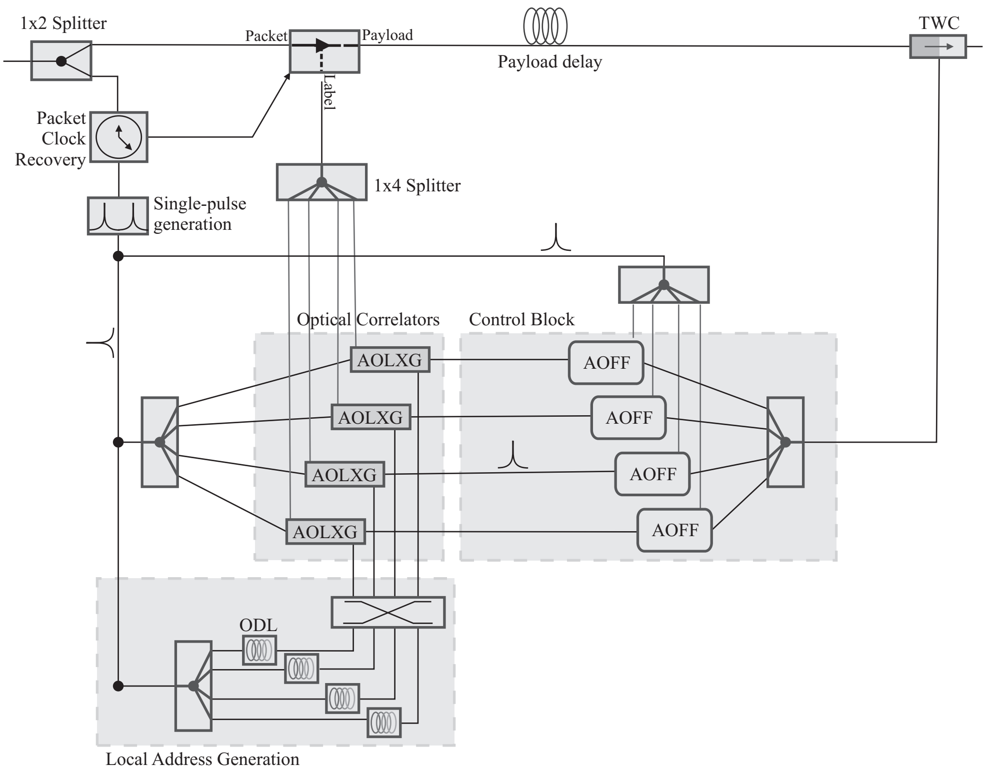


Figure 2: New Label Generation block allowing label stripping

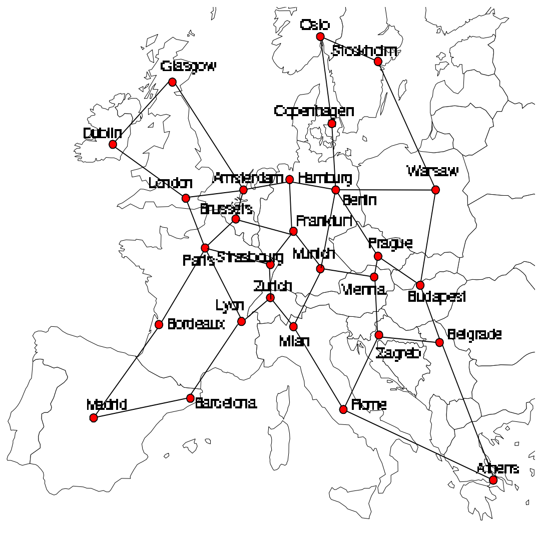


Figure 3: European Network

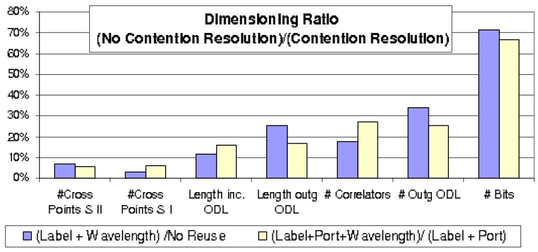


Figure 4: Dimensioning of label spaces considering Contention vs. No contention resolution ([CCPD06, Fig. 12])

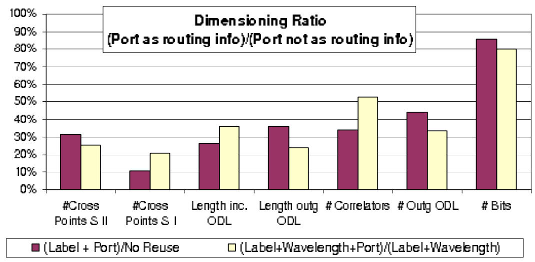


Figure 5: Dimensioning ratio II ([CCPD06, Fig. 13])

1.3.2 Adopting a *Per Fiber* Label Space

To measure the impact of the incoming port carrying routing information, the results of the label spaces types where the incoming fiber port is used as routing information (i.e. *per fiber* and *per wavelength*) is considered, and then divided by the results of similar label space types where it is not (*global* and *per wavelength color*).

It is expected that the dimensions would be smaller in case that incoming fiber port is used as routing information, as it can be seen in Fig. 5.

1.3.3 Stripping vs. Swapping

As mentioned before, although labels using stripping are short, the 'total' labels to be transported (end-to-end label) could be longer than any when labels are swapped instead. This is because in the former the end-to-end label is a concatenation of local labels. This subsection is devoted to analyze this aspect.

First of all, let us assume that an optical header contains: a label-field (comprises the forwarding information), guard bands (to separate the different information fields), and Class of Service (CoS)-field (Class of Service: to indicate the traffic priority). When labels are stripped off, the header may contain a single CoS-field for the whole header or one CoS-field for each label. Guard bands must be used between labels (in case of label stripping), between a label and a CoS-field, and to delimit the header.

Assuming that the length of a label is 8-bits (256 flows handled by an OPS) long for label swapping and 3-bits (a connectivity degree of eight in the network) long for label stripping, that both the guard bands and the CoS-field have a length of 3-bits each, and that the maximum number of hops in the network is eight; the total number of bits used in a header is shown in the Table 1.

Table 2 gives an overview of the label overhead introduced by the different switching strategies.

Assume that 50% of the payload sizes are 40-bytes long, 37.5% are 520-bytes long, and 12.5% are 1500-bytes long. The overhead caused in the network is then 3.14%, 8.13% and 12.79%. In this case, the overhead introduced by the label stripping strategy worst case is only four times bigger than that of the label swapping strategy.

2 Implementing Label Stacking in AOLS*

Enabling an AOLS block with stacking and swapping implies that the system should be able to generate one or two new labels maximum. In the case of stacking, it should be able to insert the two new labels before the packet. In the case of swapping, the system must behave as usual. Since deciding how many labels are going to be inserted depends on the content of the incoming label, to provide this flexibility is not straightforward.

The easiest way to provide two or one label insertion in the system is by allocating a fixed space for two labels even though only one is placed. The *Packet/Payload separation* circuit must be able to extract only the top label of the packet, regardless of how many they are. The remaining label, if any, should be treated as part of the payload.

Generating one or two labels can be implemented with a slight modification of the *new label generation block*, as seen in Fig. 6 [SCC+07b].

The high-intensity pulse of the AOLXG that matched the incoming label is split in two. One of the two pulses is delayed for a fraction of time equal to the duration of one label. Both pulses are switched - one after the other due to the induced delay - using the PXC in order to generate two different labels. Both labels are generated out of each pulse using the same set of ODLs mentioned before. Finally, the labels are merged into one optical stream.

In the case only one label is needed, the PXC should drop the pulse that was not delayed. In the event of popping the stack, the PXC should drop both pulses.

	Guard Bands	CoS-field	Label-field	Total
Swapping	3×3 bits	1×3 bits	1×8 bits	20 bits
Stripping (CoS per header)	10×3 bits	1×3 bits	8×3 bits	57 bits
Stripping (CoS per label)	17×3 bits	8×3 bits	8×3 bits	99 bits

Table 1: AOLS Header Sizes

Strategy	Swapping	Stripping	
		CoS per header	CoS per label
Payload\Header size (bytes)	2.5	7.125	12.375
40	5.88%	15.12%	23.63%
520	0.48%	1.35%	2.32%
1500	0.17%	0.47%	0.82%

Table 2: Network Overload due to AOLS Headers

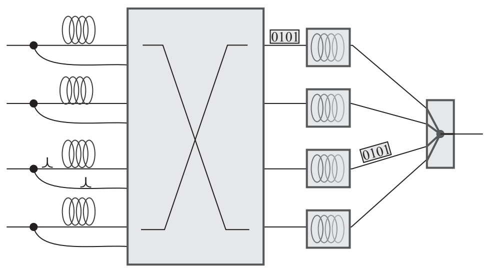


Figure 6: New Label Generation block allowing label stacking

Penultimate Hop Popping in AOLS As mentioned before, Generic Multi-Protocol Label Switching (GMPLS) allows non-labeled packets forwarding at the last hop. For this, the label stack has to be popped (hence becoming empty) in the previous hop to the last Label Switched Router (LSR). This special case is known as the Penultimate Hop Pop (PHP). When the last LSR in the path receives an unlabeled packet, the packet is treated as an IP packet (or the corresponding routing layer) in order to determine its destination.

However, because of routing in the AOLS architecture is based solely in the coded label, AOLS packets must always carry a label; including the last hop. This slight difference may increase the number of labels used in an AOLS-driven network with respect to a traditional GMPLS network.

On the other hand, if label stacking is considered, the PHP option can be used in the tunnel. Indeed, not considering the option would increase the label space, as mentioned before.

3 Label Space Size vs. MLU: An Example

In this subsection we present an example showing how the MLU affects the label space, and how this repercussion is eased when label merging and stacking are considered.

Let us consider the physical fiber topology of Fig. 7(a) with 14 nodes, in which nodes N1 and N2 are ingress and nodes N6 and N7 are egress. Let us assume that we need to route one unit of traffic from both ingresses to both egresses. More precisely, we denote by A the connection needed from N1 to

N7; B the connection from N1 to N6; C the connection from N2 to N7 and; D the connection from N2 to N6. The solutions shown in this subsection do not contemplate the possibility of splitting the demanded bandwidth across several paths.

A classical Traffic Engineering (TE) routing algorithm could aim at routing traffic such that the MLU is minimized while bounding the delay[2]. A typical TE solution for this example can be seen in Fig. 7(b). The solution has the minimum delay (or hop count) and the minimum MLU. In this case, the number of used labels equals the hop count, i.e. 22, and the MLU does not exceed two units of traffic along three links (N11-N12, N12-N13 and N13-N14).

Considering the *label merging* feature, a different routing solution leads to the usage of fewer labels. For instance, in Fig. 8(a) connection A is using a different route, so its labels can be merged with connection C from node N11 to N7. Similarly, connection B is merged with connection D. Even though the number of labels is reduced in this case down to 16, the MLU has been increased up to four units of traffic along the same three links.

Label stacking, together with label merging, gives much more options to play with. In Fig. 8(b) another solution is given to the problem reducing the label space to its minimum. In this case, 12 labels are needed while the maximum link utilization is preserved to four units of traffic along the same segment.

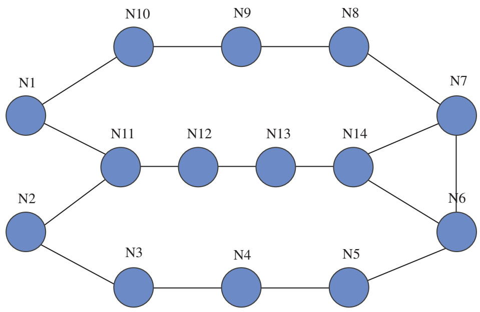
In case the MLU is limited to (or links capacities are fixed to) two units of traffic, the solutions in Fig. 8(a) and Fig. 8(b) are not feasible. However, the solution in Fig. 7(b) can use 19 labels if one tunnel is created, as seen in Fig. 9(a). In addition, in Fig. 9(b), a different routing solution that reduces the label space down to 16 can be seen as well. While the solution in Fig. 9(a) preserves the TE routing (achieving the best link utilization and delay), the solution in Fig. 9(b) increases the number of links reaching the MLU to eight but reduces even more the label space.

4 Modeling Routing & Label Space Reduction in AOLS*

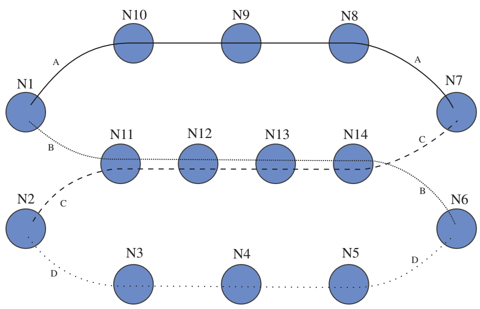
Without considering any label space reduction method (neither merging nor stacking), the total number of labels used in a network is equal to the total number of hops needed to route flows. Therefore, the longer the routes are, the more overall labels are needed.

In the previous part tackling the label space reduction problem in GMPLS flow routing was resolved by an algorithm considering several Quality of Service (QoS) metrics. As a matter of fact, routing was performed *before* - and regardless of how - labels were assigned. As a counterpart, the routing algorithm aimed at finding relatively short paths. Routing and labels binding were performed separately since it was not worth to sacrifice the QoS level for smaller label space.

[2]For simplification, we assume that hop count is equal to the delay

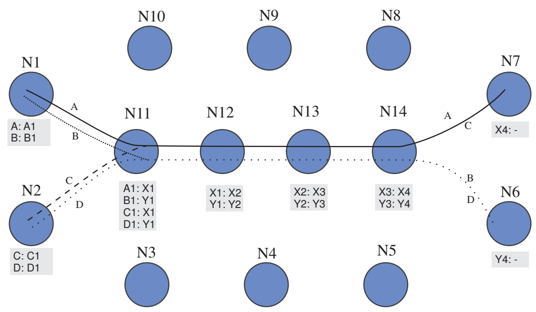


(a) Fiber Topology.

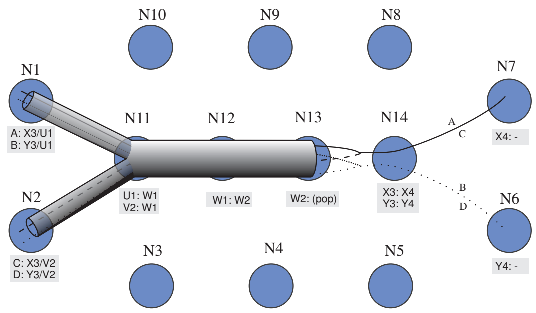


(b) 22 labels and MLU of two units of traffic along three links.

Figure 7: Routing and Traffic Engineering in AOLS.

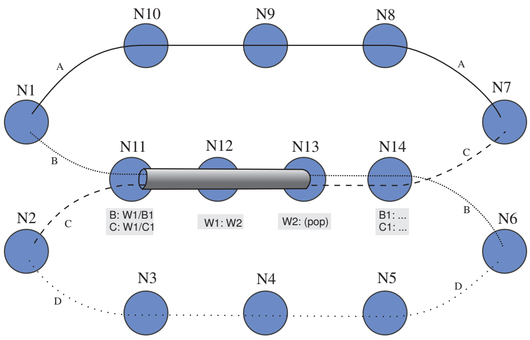


(a) 16 labels and MLU = 4u at three links (no stacking).

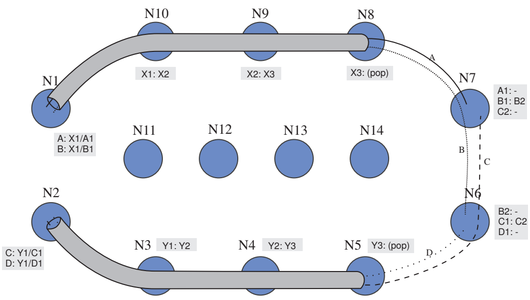


(b) 12 labels and MLU = 4u at three links.

Figure 8: Example of Link Utilization vs. Label Space Size.



(a) 19 labels and MLU = 2u at three links.



(b) 16 labels and MLU = 2u at eight links.

Figure 9: Example of Label Space Size when MLU is bounded.

In OPS using AOLS, the context is different. A slightly longer route could result in a imperceptible lower QoS, but in great CAPital EXpenditures (CAPEX) savings if label space reduction methods are considered. For instance, considering merging, employing two long "mergeable" routes may represent more CAPEX savings than employing two "non-mergeable" short ones. The overall number of used labels of the MultiPoint-to-Point Label Switched Path (MP2P) connection, in the former solution, could be less than the two Point-to-Point LSP (P2P) connections, in the later solution. Therefore, the routing problem is henceforth considered together with the label space reduction problem.

As discussed in [CCPD06], the dimensions of the AOLS-block are intrinsically related to a) the number of incoming labels, b) the number of outgoing labels, and c) the number of bits used to code optical labels in the network. These parameters are considered for analysis in this section through mathematical modeling of the problem.

At each of the following subsections, the routing problem is considered together with one of the different methods presented before in GMPLS. The complexity of the label space reduction methods used at each subsection is greater than the previous.

4.1 AOLS Routing

Initially, a simple routing model is considered without using any label space reduction method. As mentioned before, reducing label spaces in this context can be translated to routing using the shortest path: a CSPF algorithm. Although these algorithms have been extensively studied before, an ILP model is proposed with the purpose of extending it in further subsections to include label space reduction methods.

The ILP models considered throughout this section are path-based. Therefore, all feasible paths in the network are initially generated using a simple algorithm.

The following is a list of all the indexes used in the model, so far.

- $e, i, j \in \mathcal{N}$ Optical nodes in the network
- $l \in \mathcal{L}$ A generated path in the network. Sometimes l_M and l_O are used as well.
- $m \in \mathcal{M}$ A flow demand.

The parameters used in the model are:

- $R^l_{(i,j)}$ Set to 1 if link (i, j) is used by the generated path l
- S^l_j Set to 1 if node j is the source of the generated path l
- D^l_j Set to 1 if node j is the destination of the generated path l
- C^m Set to the bandwidth demand of m

- Δ_j^m Set to 1 if node j is the source of demand m, -1 if it is its destination, and 0 otherwise.
- F Set to the number of wavelengths per fiber
- W Set to the capacity of each wavelength
- L The maximum number of bits used to encode labels.

The solely variable used by this model is $\alpha^{l_M,m}$, set to 1 when demand m is routed using path l_M.

The objective function is to reduce the number of labels (or hops in this context) of every routed demand.

$$\min \sum_{l,m} R^l \cdot \alpha^{l,m}, \quad \text{where} \tag{1}$$

R^l is the number of hops used by route l, i.e. $\sum_{(i,j)} R^l_{(i,j)}$

Subject to:

$$\sum_{l|S_j^l=1} \alpha^{l,m} = 1, \qquad \forall j, m \text{ s.t. } \Delta_j^m = 1 \tag{2}$$

$$\sum_{l|D_j^l=1} \alpha^{l,m} = 1, \qquad \forall j, m \text{ s.t. } \Delta_j^m = -1 \tag{3}$$

$$\sum_{l} \alpha^{l,m} = 1, \qquad \forall m \tag{4}$$

$$\sum_{l,m|R^l_{(i,j)}=1} C^m \cdot \alpha^{l,m} \leq F \times W, \qquad \forall i, j \tag{5}$$

The first three equations assure that the path used to route a flow demand is consistent with the source and destination nodes of the demand. The last equation limits the capacity that can be used in every fiber link.

The labels length is bounded with the expressions.

$$\sum_{i,l,m|R^l_{(i,j)}=1} \alpha^{l,m} \leq 2^L, \qquad \forall j \tag{6}$$

$$\sum_{i,l,m|R^l_{(j,i)}=1} \alpha^{l,m} \leq 2^L, \qquad \forall j \tag{7}$$

4.2 Aggregating AOLS flows

The number of labels can be drastically reduced if it is noticed that several demands between a pair of nodes are routed, by the previous model, through the same paths. Therefore, it is normal to consider that all these demands following the same end-to-end paths can be labeled equally. This is, assign one label per each different used path in the network regardless of how many demands are using it. It can be assumed that differentiating the flows is performed outside the optical domain, once the signal is given to the underlying electronic switch or local area network, for instance.

To model this, a new variable is introduced: δ^l, set to 1 if path l is used to route any demand.

The objective function is:

$$\min \sum_{(i,j),l} R^l_{(i,j)} \cdot \delta^l \tag{8}$$

subject to all the constraints previously presented plus an additional one linking the two variables $\alpha^{l,m}$ and δ^l:

$$\alpha^{l,m} - \delta^l \leq 0, \qquad \forall l, m \tag{9}$$

Since labels are assigned to paths instead than to demands, the expression bounding the labels length shall be modified.

$$\sum_{i,l|R^l_{(i,j)}=1} \delta^l \leq 2^L, \qquad \forall j \tag{10}$$

$$\sum_{i,l|R^l_{(j,i)}=1} \delta^l \leq 2^L, \qquad \forall j \tag{11}$$

4.3 Label Merging in AOLS

Label merging in MPLS was deeply analyzed in [SFM07]. A polynomial-time algorithm solves the label bindings if the routes are given. However, since the demand routes are considered as not fixed in AOLS, an ILP is needed to perform labels merging together with routing [SCC+07a].

In the last objective function the decision variable (δ^l) was multiplied by a constant factor ($\sum_{(i,j)} R^l_{(i,j)}$) in order to calculate the number of used labels per each employed path l. This is because the number of labels used by a path is fixed. In order to perform label merging in this model, it is considered that not all links of a path use labels. When a link of a path is not using a label it is because another "mergeable" path can share its label with the former path.

In order to decide which paths would use a label in a link, a new variable is needed in the model: $r^l_{(i,j)}$. The variable is set to 1 when a label is employed at link (i,j) for path l. As mentioned in previous subsection, it should be recalled that labels are assigned to paths instead than to demands.

The objective function is therefore:

$$\min \sum_{(i,j),l} r^l_{(i,j)} \tag{12}$$

It is clear that the variable takes the value of zero (0) in all its links when the path is not used. However, such constraint can be reformulated in order to restrict more the search space as follows.

If the paths that are going to be used could be known in advance, determining how labels can be merged for those paths is a deterministic polynomial-time decision [SFM07]. Given a set of paths with the same egress, the same label can be allocated in a particular link to a set of paths if the paths follow exactly the same route from the link to the egress LSR. It should be noted that given a particular link and a set of paths with the same egress LSR, there could be several groups of paths that cannot use the same label. For instance, in Fig. 10, the link $N11 \rightarrow N10$ has two groups of paths that cannot be merged even though all the paths have the same egress node. The first group is conformed by paths B and C, and the second by paths D and E.

Since the paths that are going to be used to route the demands are not known in advance, the ILP must know how to merge any of the feasible paths to conform a MP2P connection in the network. For this, a new parameter is introduced in the ILP: $M^{e,l}_{(i,j)}$. Given a link (i,j) and an optical node e (identifying a MP2P connection in the network), the value of $M^{e,l}_{(i,j)}$ is the same for all the paths l that can use the same label in the link (i,j). In other words, the parameter $M^{e,l}_{(i,j)}$ groups (with its value) the paths l that can be merged in a particular link (i,j) for a MP2P rooted at e.

Computing this parameter is performed in polynomial time using an algorithm similar to the Full Label Merging algorithm [SFM07], if all possible paths in the network are considered. Given a link (i,j) and a root e, the values of $M^{e,l}_{(i,j)}$ are assigned consecutively to each set of mergeable paths l. For notation simplicity, let us assume that $\hat{M}^{e}_{(i,j)}$ is the maximum value (i.e. group number) in a particular link of a MP2P connection.

Considering a link (i,j) of a MP2P connection with root e, the number of labels used should be (at least) one for all the paths belonging to the same group. This is expressed with the following constraint.

$$\sum_{l|M^{e,l}_{(i,j)}=k} \left(\delta^l - \|\mathcal{L}\| \cdot r^l_{(i,j)}\right) \leq 0, \qquad \forall e,i,j,k \in \mathbb{N}, 1 \leq k \leq \hat{M}^{e}_{(i,j)} \tag{13}$$

In the previous constraint, for a given group k, if any of the paths δ^l is used, labels in that link ($r^l_{(i,j)}$) are needed. Note that the rate between all the used paths in group k ($\sum_{l|M^{e,l}_{(i,j)}=k} \delta^l$) and the number of labels used in link (i,j) for group k ($\sum_{l|M^{e,l}_{(i,j)}=k} r^l_{(i,j)}$) is $\|\mathcal{L}\|$. Since

$$\|\mathcal{L}\| \cdot r^l_{(i,j)} \geq \sum_{l'} \delta^{l'},$$

for the set of paths of particular group in a link, an optimal solution contains

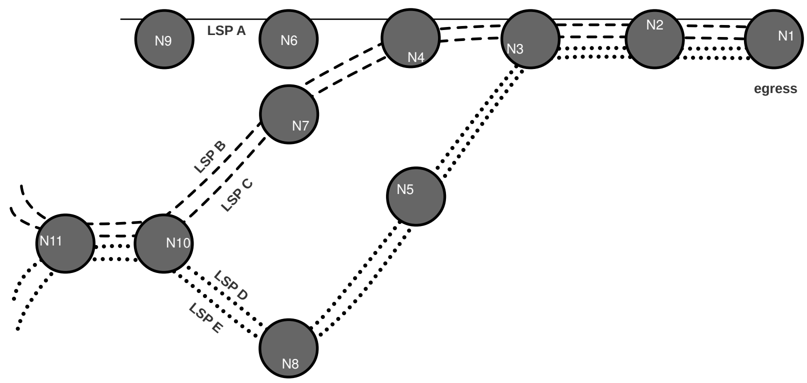


Figure 10: MP2P Scenario with five LSPs with the same egress LSR $N1$. There may exist more than one way to perform merging since $N11 \rightarrow N10$ is crossed by more than two diverging LSPs.

only one label ($r^l_{(i,j)}$) in the group regardless of how many paths of that group are being used.

The number of incoming and outgoing labels is restricted with the expressions.

$$\sum_{i,l} r^l_{(i,j)} \leq 2^L, \qquad \forall j \tag{14}$$

$$\sum_{i,l} r^l_{(j,i)} \leq 2^L, \qquad \forall j \tag{15}$$

4.4 AMT in AOLS

In this subsection a model considering Asymmetric Merged Tunnel (AMT) with full label merging and routing is proposed.

In this model, there are two types of paths that need to be differentiated. While the first type is the same used so far to route the demands, the second ones is the path used by a branch of an AMT. For this purpose, the paths used to route demands are denoted with the index l_M and the paths used by a branch of an AMT are denoted with the index l_T.

In order to handle the AMTs, two new variables are needed: γ^{l_T} and β^{l_M,l_T}. Variable γ^{l_T} is set to 1 when an AMT uses the path l_T as a branch to perform stacking on a set of demands path. Variable β^{l_M,l_T} is set to 1 when the branch following path l_T of an AMT performs stacking over demands routed through the path l_M. In other words, variable γ^{l_T} "summarizes" variable β^{l_M,l_T}.

The number of labels that are saved by using the branch l_T with the path l_M is gathered by the parameter Δ^{l_M,l_T}.

Similarly to the previous model, the variable $r^{l_T}_{(i,j)}$ is used to determine which links are using labels in an AMT branch.

The objective function is formulated as follows.

$$\min \sum_{(i,j),l_M} R^{l_M}_{(i,j)} \cdot \delta^{l_M} - \sum_{l_M,l_T} \Delta^{l_M,l_T} \cdot \beta^{l_M,l_T} + \sum_{(i,j),l_T|D^{l_T}_j=0} r^{l_T}_{(i,j)} \tag{16}$$

The expression adds the number of total labels without AMTs (as consequence of routing), substracts the number of labels saved because of AMTs and adds the number of labels used by using AMTs.

The MP2P constraint of previous subsection is replaced by a similar one concerning AMT.

$$\sum_{l_T|M^{e,l_T}_{(i,j)}=k} \left(\gamma^{l_T} - \|\mathcal{L}\| \cdot r^{l_T}_{(i,j)}\right) \leq 0, \qquad \forall e,i,j,k \in \mathbb{N}, 1 \leq k \leq \hat{M}^e_{(i,j)} \tag{17}$$

The number of AMTs stacking one path per link is constrained as.

$$\sum_{l_T|R^{l_T}_{(i,j)}=1} \beta^{l_M,l_T} \leq 1, \qquad \forall i,j,l_M|R^{l_M}_{(i,j)} = 1 \tag{18}$$

The number of incoming and outgoing labels is bounded with the following two expressions.

$$\sum_{i,l_M|R^{l_M}_{(i,j)}=1} \left(\delta^{l_M} - \sum_{l_T|R^{l_T}_{(i,j)}=1} \beta^{l_M,l_T}\right) + \sum_{i,l_T|D^{l_T}_j=0} r^{l_T}_{(i,j)} \leq 2^L, \qquad \forall j \tag{19}$$

$$\sum_{i,l_M|R^{l_M}_{(j,i)}=1} \left(\delta^{l_M} - \sum_{l_T|R^{l_T}_{(j,i)}=1} \beta^{l_M,l_T}\right) + \sum_{i,l_T|D^{l_T}_j=0} r^{l_T}_{(j,i)} \leq 2^L, \qquad \forall j \tag{20}$$

The new variables are linked using the following set of constraints.

$$\beta^{l_M,l_T} - \delta^{l_M} \leq 0, \qquad \forall l_M, l_T \tag{21}$$

$$\beta^{l_M,l_T} - \gamma^{l_T} \leq 0, \qquad \forall l_M, l_T \tag{22}$$

5 Solving the Routing and Label Space Reduction Problem using Heuristics*

The problem is solved in two-steps by two separate algorithms proposed in this section. The first algorithm routes a traffic demand matrix aiming at setting up paths such that they share the maximum number of links. Once all demands (if possible) are routed, the second algorithm creates a set of tunnels reducing the label space.

5.1 The Path-Interfering Routing Algorithm (PIRA)

As routing solution, we propose a modification of the Constraint Shortest-Path algorithm (CSPF). The modification consists in how the weight of links is computed. Traditionally, the weight of a physical link is set to its (propagation) delay or proportional to its available bandwidth. Here, we propose a new dynamic weight that aims at favoring links using more labels. Our intention is that, after many iterations of the routing algorithm, there are going to be segments in the network that are highly loaded with paths. These segments will later cause a high label space reduction when tunnels are placed to cover them.

Given a network $G^*(V, E^*)$ and a set of paths P, we extend G^* to a *directed multigraph* $G(V, E)$ as follows.

Every node and link in G^* are also considered in G. We name these links: *physical links*. The bandwidth of a physical link in G is set to the available bandwidth of its corresponding link in G^*. In addition, its weight is fixed to one.

Given a pair of non-adjacent nodes $i, j \in V$ and the set of paths P, it is worth noting that:

- There could be several paths that are forwarded through the same segment from i to j.
- Considering the set of paths P, there could be more than one segment that forwards information between i and j.

Regardless of which paths they belong to, we denote $s^{(k)}_{i \to j}$ the k-th different segment from i to j considering P. We create one different link from i to j in G for every $s^{(k)}_{i \to j}$. We name these links: *induced links*. The bandwidth of an induced link in G is set to the minimal available bandwidth of the links in G^* conforming its corresponding segment. In addition, its weight is set to

$$\frac{|s^{(k)}_{i \to j}| - 1}{|P(s^{(k)}_{i \to j})| + 1}, \quad \text{where}$$

$|s^{(k)}_{i \to j}|$ is the length of the segment and $|P(s^{(k)}_{i \to j})|$ is the number of paths in P that traverse the segment. The idea behind the weight is that a new path "pays", at every hop, only for the share of the label that it uses. One (1) is subtracted from the numerator because the last hop of $s^{(k)}_{i \to j}$ never incurs in labels reduction [SSFM07]. One (1) is added to the denominator in order to consider the new path as well.

When CSPF takes an induced link to route a demand, the link should be interpreted instead as the set of physical links in G^* representing it. If a cycle is created in the physical links, it is broken. It is worth noting that, even though the number of links contemplated increases, the complexity of the CSPF-procedure should not be worse than that run with a full mesh graph.

5.2 The Most-Profitable Tunnel First Algorithm (MPTF)

We consider that path routes were already computed. Taking into account these path routes, we compute the set of tunnels to consider as described in [SSFM07].

It is worth noting that the number of labels that ϕ can reduce by covering α is $|\alpha \cap \phi| - 1$, however ϕ needs $|\phi| - 1$ labels despite the number of paths it covers. This gives us an idea of the *profit* gained by setting up a tunnel. More concretely, the profit of a tunnel is the number of labels that it reduces by covering all the non-covered segments of the paths.

Note that our metric (i.e. the profit) considers the "savings" due both covered length and covered broadness, gathering in one metric the best of both LSF and MCSF.

At each iteration, a tunnel - name it ϕ - with the *best profit* is always selected. Once a tunnel is selected, all the non-covered segments of paths that ϕ can cover - name them $P(\phi)$ - are marked as covered. These covered segments will not be considered in further iterations. The profit of all remaining tunnels must be updated. The algorithm iterates until no tunnel remains.

At the beginning of the algorithm, since all paths are completely uncovered, the profit of a tunnel ϕ is set to:

$$\Pi(\phi) \leftarrow 1 - |\phi| + \sum_{\alpha \in P(\phi)} (|\alpha \cap \phi| - 1)$$

Before the next iteration takes place, the profit of all the others tunnels θ overlapping with ϕ is updated accordingly to:

$$\Pi(\theta) \leftarrow \Pi(\theta) + |\phi \cap \theta| \cdot \delta_{\phi,\theta} - \sum_{\alpha \in P(\phi)} (|\alpha \cap \theta| - 1), \quad \text{where}$$

$\delta_{\phi,\theta}$ is set to one (1) if tunnels ϕ and θ end at the same node, zero (0) otherwise.

Neither ϕ nor any other tunnel $\phi' \subset \phi$ are considered any more in further iterations. The complexity of the algorithm is bounded by $O(t \cdot \log t)$, where t is the number of feasible tunnels.

6 Simulation Experiments

In this section, we present a set of simulations that shows the discussed trade-off here.

6.1 Heuristic Performance

The European network consisting of 37 nodes is used in our simulation experiments, see Fig. 11. The link capacity is varied from 100 units to 3000 units of traffic, creating different simulation scenarios limiting the MLU. We use the term link capacity and MLU to denote the same throughout our analysis.

The first 30% of the nodes with lowest connectivity degree are selected as edge (ingress/egress) routers. For each pair of edge routers, a number of X, Y and Z demands of one, three and 12 units of traffic is randomly generated. X, Y and Z follow an uniform distribution with parameters [0-20], [0-12] and [0-4] respectively. As a result, we generate an average of 200 demands with average demanded bandwidth of 6864 units of traffic.

In this subsection we tested several combinations of routing heuristics with label space reduction heuristics. Namely, we considered as routing heuristics: CSPF and PIRA (proposed here). As for label space reduction heuristics, we considered: MultiPoint-to-Point [SMY00, AT03] (MP2P, i.e. label merging without stacking), LSF, MCSF and, MPTF (proposed here).

In total, eight routing-tunneling solutions are contemplated. In Fig. 12(a), simulation results show that the best label space reduction is achieved by PIRA-MPTF when the MLU limit is high, and CSPF-MPTF when the MLU is low. In general, PIRA obtains better reductions when used with high MLUs and with any stacking heuristic (i.e. LSF, MCSF and MPTF). In the same way, MPTF obtains the best reduction regardless of the routing heuristic.

The use of the stack reduces the label space four times (CSPF-MP2P vs. CSPF-MPTF) in average when the capacity of the links is just enough to route traffic (around 1000 units of traffic), and almost six times if the capacity is doubled (CSPF-MP2P vs. PIRA-MPTF).

It is worth noticing that the number of labels increases at the beginning, when the MLU limit is low. This behavior is explained by the fact that the routing heuristics (either CSPF or PIRA) employ more, and longer, paths to accommodate the traffic in these scenarios, in order to avoid the violation of the MLU limit. In case of PIRA, this behavior is stronger and particulary emphasized (see peaks in figure) when the MLU limit is set to 800 and 1400 units of traffic.

As expected, PIRA creates more link bottlenecks than CSPF in order to provide more tunneling possibilities, see Fig. 12(b). Therefore, the usage of PIRA is advisable when the minimum spare capacity of the links doubles the MLU. In our simulations, we noticed that the usage of PIRA in this circumstances profits with a 30% in the label space reduction.

We proceed to focus now in the particular routing solutions when the capacity of the links is high. While the MLU utilization of CSPF is 934 units of traffic, PIRA's is 2236; this is 2.5 times more. However, we notice that this case occurs in few links. In Fig. 13 we show the number of links in PIRA whose used capacity is above (or below) a given percentage of the MLU of CSPF (934 units of traffic). For instance, there are six links in PIRA that are using between 50% and 75% *more* capacity than the minimum MLU considering CSPF routing. It turns out that while 25 links (out of 114) requires a higher capacity, 74 are not used by PIRA (16 of them are not used by CSPF either). This suggests the idea of either: *a*) rewiring the network (instead of increasing the existing links capacities) or, *b*) employing the unused links to create lightpaths providing the extended capacity to overused links. These ideas will be studied in further contributions.

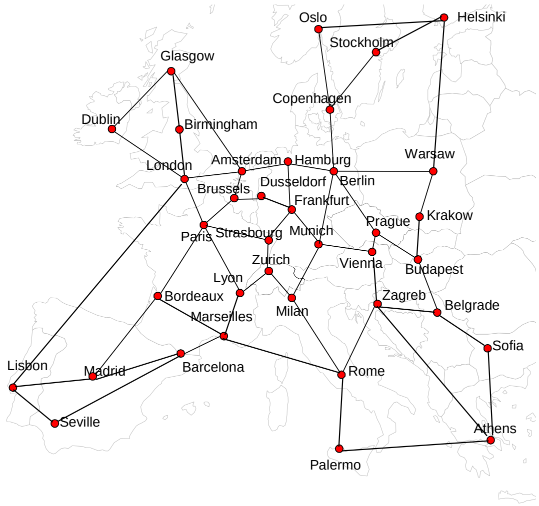
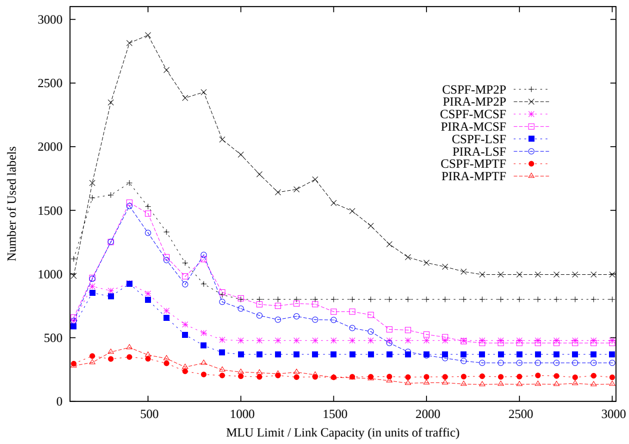


Figure 11: European network with 37 nodes.

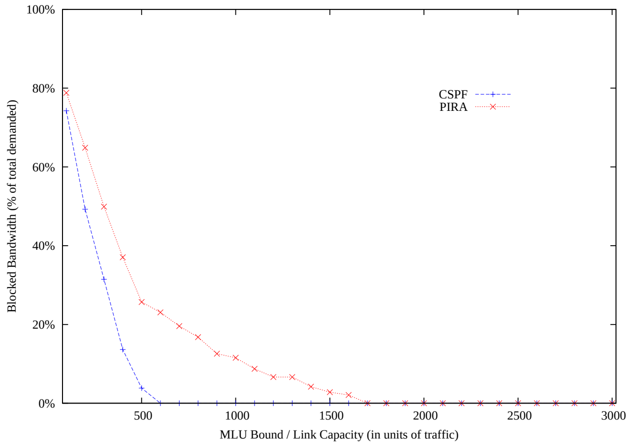
We compare now the best solution without stacking (CSPF-MP2P) with the best solution using the stack (PIRA-MPTF). The maximum number of labels that CSPF-MP2P uses is 20. This makes all AOLS-blocks to be sized for coding labels of five bits long. By using stacking (PIRA-MPTF), the maximum number of labels becomes 11, making the AOLS-blocks to be sized for coding labels of only four bits long. In Fig. 14 we show the number of links that are using a number of labels within a given range. It shows that 11 links are causing the five bits long labels without stacking. However, using the stack, only four links are forbidding us from using three bits long labels. Even though we did not optimize the maximum number of labels per link, it is not difficult to see that rerouting the traffic in the three links of PIRA-MPTF would be easier than rerouting the traffic of the 11 links in CSPF-MP2P in order to reduce one bit the label encoding size.

Considering the header in [RKM$^+$05], we compute the overhead due the coding of the extra label. In the case of no label stacking, the five bits long label yields to a 17 bits header. In the case of label stacking, the four bits long labels yield to a header of 23 bits. Assuming a classical packet distribution (as in [CCPD06]), the average overhead caused in the traffic of the network is just 0.18% more due stacking.

It is worth noting that if *label stripping* using CSPF is considered, packet header must code eight labels (minimum distance in CSPF paths) in the header and the network must be able to handle 102 labels (the number of used links by CSPF). If PIRA is considered, packet header must code 17 labels, and the network must be able to handle 40 labels. However, the traffic overhead is 2.02% and 4.5% more, respectively. Therefore, label stacking offers a better trade-off.



(a) Label Space Size vs. Link Capacity.



(b) Blocked Bandwidth.

Figure 12: Heuristics Overall Performance.

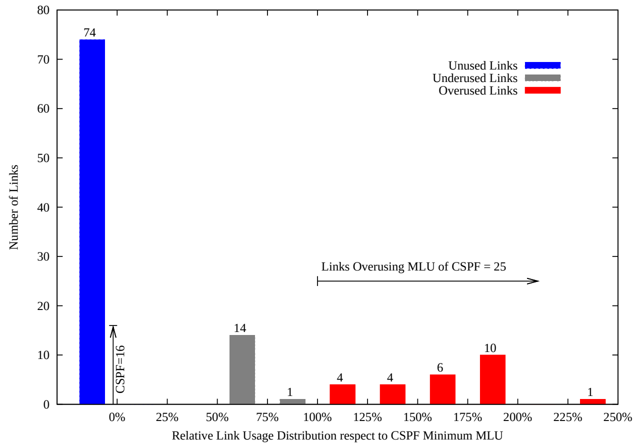


Figure 13: Distribution of overused link capacities of PIRA respect to CSPF MLU.

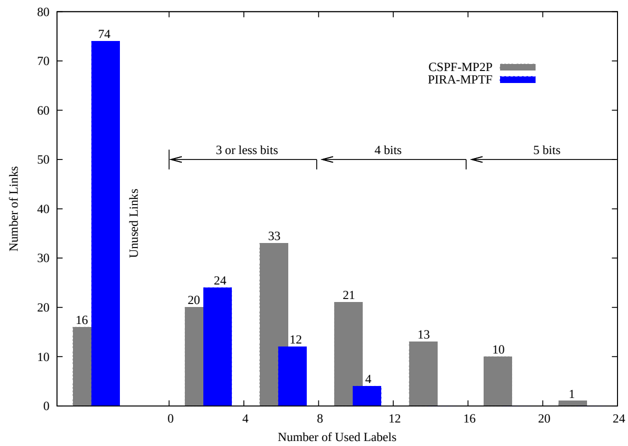


Figure 14: Distribution of the label space link-by-link using PIRA-MPTF.

6.2 How Far From Optimum?

Each one of the two heuristics presented here affect the overall optimality of the solutions. Therefore, we compare our results with the optimal in two steps:

6.2.1 Minimize Labels considering CSPF/PIRA Routes

We route the traffic using CSPF and PIRA and then we relax the proposed ILP so it computes the best label space reduction using CSPF/PIRA paths. We consider the network scenario in which the MLU limit is the highest. By solving the relaxed ILP using the CSPF routes, the minimum number of used labels considering stacking is 180. Considering PIRA, the minimum number of used labels decreases down to 109. In this way, we can claim that: *a*) PIRA-routing leads to paths aiming at better label space reduction in general and, *b*) MPTF performs 21.11% far from its optimal value considering PIRA. Henceforth, we only consider PIRA-MPTF.

6.2.2 Minimize Labels with Variable Routes

We solve the complete ILP model proposed in §4.4 with a smaller network shown in Fig. 15. Edge nodes and demands are generated following the same parameters of the European network. The MLU is set to the double of the minimum needed by CSPF. We found that PIRA-MPTF label space size is 26.7% more than the optimal. A similar test to the previous one exposes that 68% of the error is caused by the selection of the routes. The reason is the tendency of PIRA for using more paths than the optimal. This behavior is mainly explained by the link congestions caused by PIRA and by the usage of long routes.

7 Remarks

The label space reduction problem in AOLS was tackled. First of all, an overview of the state of the art in AOLS is given. Second, an extension of the current architecture, given in the LASAGNE project, is made in order to allow the stacking (pushing and popping) of one label completely optically.

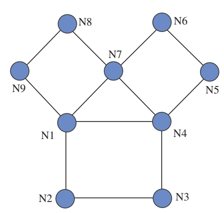


Figure 15: Network simulated for ILP

Due the cost expenditures of handling a label in AOLS, the routing problem is considered together with the label space reduction problem. So, the purpose is to route traffic and create tunnels in such a way that the minimum number of labels is used. As a consequence, it was foresen that the routing solution would create bottlenecks. Therefore, as a trade-off metric, the maximum link utilization in the network is analyzed as well.

The results show that, due the flexibility of routing, MP2P achieves greater reduction ratios than without such flexibility. AMTs, on the other hand, are not able to gain much. However, AMTs use better the spare capacity in the links than MP2Ps in order to decrease the label space.

References

[AT03] David Applegate and Michel Thorup. Load optimal MPLS routing with N+M labels. In *Proc. IEEE INFOCOM 2003*, pages 555–565, 2003.

[BPY+02] C. Bintjas, N. Pleros, K. Yiannopoulos, G. Theophilopoulos, M. Kalyvas, H. Avramopoulos, and G. Guekos. All-optical packet address and payload separation. *IEEE Photon. Technol. Lett.*, 14:1728–1730, 2002.

[CCPD06] Ruth Van Caenegem, Didier Colle, Mario Pickavet, and Piet Demeester. Benefits of label stripping compared to label swapping from the point of node dimensioning. *Photonic Network Communications Journal*, 12(3):227–244, December 2006.

[MCL+02] S. De Maesschalck, Didier Colle, I. Lievens, Mario Pickavet, and Pitier Demeester. Pan-european optical transport networks: an availability-based comparison. *Photonic Network Communications Journal*, 5(3):203–225, May 2002.

[MRM+02] J.M. Martinez, F. Ramos, J. Marti, J. Herrera, and R. Llorente. All-optical N-bit XOR gate with feedback for optical packet header processing. In *Eur. Conf. Optical Communication (ECOC)*, volume 3, 2002. Poster session.

[RKM+05] Francisco Ramos, E. Kehayas, Jose M Martinez, R. Clavero, Javier Marti, L. Stampoulidis, D. Tsiokos, H. Avramopoulos, J. Zhang, P. V. Holm-Nielsen, N. Chi, P. Jeppesen, N. Yan, I. Tafur Monroy, A.M.J. Koonen, M.T. Hill, Y. Liu, H.J.S. Dorren, Ruth Van Caenegem, Didier Colle, Mario Pickavet, and B. Riposati. IST-LASAGNE: Towards all-optical label swapping employing optical logic gates and optical flip-flops. *IEEE J. Sel. Areas Commun.*, 23(10):2993–3011, October 2005.

[SCC+07a] Fernando Solano, Ruth Van Caenegem, Didier Colle, Jose Marzo, and Ramon Fabregat. Label merging in all-optical label swapping networks. In *Proc. Workshop in G/MPLS Networks*, April 2007. Invited Paper.

[SCC+07b] Fernando Solano, Ruth Van Caenegem, Didier Colle, Mario Pickavet, Jose Marzo, and Ramon Fabregat. Routing and label stacking in all-optical label swapping networks. In *Proc. European Conference in Networks and Optical Communications (NOC 2007)*, June 2007. Invited Paper.

[SFM07] Fernando Solano, Ramon Fabregat, and Jose Marzo. On optimal computation of MPLS label binding for MultiPoint-to-Point connections. *IEEE Trans. Commun.*, 2007. Accepted for publication.

[SMY00] Hiroyuki Saito, Yasuhiro Miyao, and Makiko Yoshida. Traffic engineering using multiple MultiPoint-to-Point LSPs. In *Proc. IEEE INFOCOM 2000*, pages 894–901, 2000.

[SSFM07] Fernando Solano, Thomas Stidsen, Ramon Fabregat, and Jose Marzo. Label space reduction in MPLS networks: How much can one label do? *IEEE/ACM Trans. Netw.*, 2007. In 2nd review.

Agència de Gestió d'Ajuts Universitaris i de Recerca

L'AGAUR inclourà a RECERCAT la memòria justificativa en versió digital. Per aquest motiu, us demanem el vostre consentiment per introduir a RECERCAT el contingut científic de la vostra justificació de recerca. Si esteu d'acord amb aquesta proposta, cal que llegiu la **llicència Creative Commons** i ens envieu el document que us adjuntem imprès, omplert i signat a:

Programa de documentació científica
Agència de Gestió d'Ajuts Universitaris i de Recerca
Via Laietana, 28 2na planta
08003 Barcelona

RECERCAT

La persona sotasignant, , en qualitat de beneficiària de l'ajut de l'Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR), amb núm d'expedient, manifesta el seu consentiment exprés per tal que la justificació presentada en el marc de l'esmentat ajut sigui introduïda al dipòsit digital RECERCAT del Consorci de Biblioteques Universitàries de Catalunya, d'acord amb els termes especificats a la llicència Creative Commons de Reconeixement-No-Comercial-SenseObra Derivada.

Aquesta llicència estableix que es permet copiar, distribuir i comunicar públicament l'obra sempre que se'n citi l'autor original i la institució que l'empara i no se'n faci cap ús amb finalitats comercials ni obra derivada. Per obra derivada s'entén aquell document que ha estat editat, traduït, combinat amb materials de tercers, canviat de format, o modificat de qualsevol altra forma.

☒ Sí, estic d'acord amb el text anterior.
☐ No, no estic d'acord.

Barcelona, 5 de Septiembre de 2007

Signatura
DNI

Aquelles persones o institucions que ho requereixin, amb una adequada argumentació, podran sol·licitar, si escau, un retràs màxim de dos anys per introduir la memòria justificativa a Recercat.

Generalitat de Catalunya
Departament d'Innovació, Universitats i Empresa