Matheuristic with machine learning-based prediction for software-defined mobile metro-core networks

Rodolfo Alvizu, Sebastian Troia, Guido Maier and Achille Pattavina

***Abstract*—In general, humans follow a routine with highly predictable daily movements. For instance, we commute from home to work in a daily basis, and visit a selected set of places for commercial and recreational purposes during the night and weekends. The use of mobile phones increases when commuting in public transportation, during lunch break and at night. Such regular behavior creates predictable spatio-temporal fluctuations of traffic patterns. In this paper we introduce a matheuristic for dynamic optical routing that can be implemented as an application into a software-defined mobile carrier network. We use machine learning to predict tidal traffic variations in a mobile metro-core network, that allows to solve Off-line mixed integer linear programming instances of an optical routing (and wavelength) assignment optimization problem. The optimal results are used to favour near optimal On-line routing decisions. Results demonstrate the effectiveness of our On-line methodology, with results that match almost perfectly the behavior of a network that performs optical routing reconfiguration with a perfect, oracle-like, traffic prediction and the solution of an optimization problem.**

***Index terms*—Network optimization, Energy efficiency, mobile metro-core network, dynamic optical routing, software-defined networking, machine learning, prediction, artificial neural networks.**

I. Introduction

TRANSPORT networks often suffer from resource inefficiency due to over-provisioning. It's a common practice to perform static resource allocation based on the peak-hour demand, because the current operational processes used by the network operators are too slow to dynamically allocate the resources following the daily demand variations. Over-provisioning leads to poor energy efficiency and high operational expenses (OpEx), as the resources are sub-utilized outside of the peak hour. Moreover, as the peak-hour to average demand ratio continue to increase [1], the static resource allocation lead to higher and unnecessary OpEx and capital expenditures (CapEx).

From a survey done to 47 Communication Service Providers (CSP)s, 2017 is the year when most the CSPs will experiencing the decline of revenue-per-bit below the cost-per-bit [2]. Luckily, there are new technologies Telecom operators can nowadays adopt to introduce automation and real-time flexibility in their manually configured and static network systems: Software Defined Networking (SDN) [3] and Network Function Virtualization (NFV). These technologies are able to provide programmability and agility to the Telecom industry [4], reducing costs and improving profits, while meeting the requirements of their customers.

Nevertheless, due to the highly predictable daily movements of large populations of citizens in urban areas [5], mobile data traffic[1] exhibits repetitive patterns with spatio-temporal variations. This behavior has been recently compared to the rise and fall of the sea levels, known as tides. Thus, it was called the tidal traffic scenario [6]. Spatio-temporal traffic variations were first studied for planning and energy efficient operation of cellular networks [7]. We will show that the dynamic optimization based on predictive models is very effective in achieving large OpEx savings in a tidal-traffic context.

One of the techniques enabled by SDN is the use of dynamic resource allocation to follow the variation of traffic demand in the network. Its has been recently observed that in the case of Mobile cellular Networks such variations happens not just in time, but also in space, due to the mobility of the users.

In this paper we introduce a matheuristic (i.e. interoperation of heuristic and mathematical programming) for dynamic network optimization. Traffic is predicted on different spatial locations using a machine-learning algorithm: predicted traffic-demand is then used to optimize the network at various hours during the day, so to adapt resource occupation to the actual traffic volume. The time required by optimization is not an issue, since prediction allows to start performing optimization computation in advance (Off-line). Such feature makes the solution reported suitable to be implemented as an SDN application for 5G scenarios. We have chosen as use-case the metro infrastructure of a mobile operator, and in particular the objective of the resource optimization is the optical metro network used as backbone for the mobile service. The matheuristic here proposed improves the methods described in [8], [9] by moving complex calculations to an Off-line phase thanks to the use of traffic prediction. In consequence, the new Algorithm introduces a very light overhead to make (On-line) near-optimal routing decisions.

This work was supported by the EU FP7 IRSES MobileCloud Project under Grant 612212.

R. Alvizu is with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milan 20133, Italy, and also with the Department of Electronics and Circuits, Simon Bolivar University, Caracas 89000, Venezuela (e-mail: rodolfoenrique.alvizu@polimi.it).

S. Troia, G. Maier, and A. Pattavina are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milan 20133, Italy (e-mail: sebastian.troia@polimi.it; guido.maier@polimi.it; achille.pattavina@polimi.it).

[1]Mobile data traffic refers today to data traffic over cellular networks such as 2G, 3G and 4G radio systems. It is important to notice that at present, mobile data traffic does not include the traffic from wireless systems such as Wi-Fi, which provides a "wireless" access to a fixed Internet connectivity. 5G will be the first mobile technology taking into account the convergence with fixed wireless (fixed and mobile convergence). However, in this paper we will not consider traffic from Wi-Fi access points, as these sources were not present into the dataset we have used.

The paper is organized as follows: section II presents related works. Section III provides a primer on traffic prediction. Then in section IV introduces the matheuristic for dynamic optical routing. Section V describes the proposed traffic prediction method. Section VI defines the Mobile Carrier Network (MCN), and describes the dataset that was used. Numerical results are reported in section VII. Finally, a discussion on open issues and the conclusions are presented in sections VIII and IX, respectively.

II. Related works

In [5], a human mobility analysis of a MCN shows that the user mobility is highly predictable (from 80% to 93%), due to the inherent regularity of human behavior.

The relation between human mobility and time-dependent traffic fluctuations in the network has been highlighted in Ref. [10], a study about Wi-Fi networks[2]. This study however neglected the spatial variations.

The energy-efficiency performance of networks that can adapt to traffic load variations in time was analyzed in [11]. Since base stations are the most power-hungry devices in MCN architectures, dynamic resource allocation and energy efficiency efforts in MCN are mainly focused on the RAN (see Fig. 4) [12], [13]. Also in these works on resource allocation and energy efficiency optimization, only the temporal fluctuation of an overall traffic demand is considered. However, such homogeneous traffic matrix is far from the behavior of traffic load in a metropolitan area.

Tidal traffic may create spatio-temporal variations that follow a regular pattern given by the human commutation from residential to working areas. Tidal traffic was first considered for optimization of RANs, considering a small cluster of MCN cell sites [7]. Then, in [14] tidal traffic was used to proposed energy efficient management for passive optical networks. Recently, tidal traffic effect was considered in [6], [15] to enhance energy efficiency in metropolitan optical networks. A limitation of these works, is that they assumed mainly two basic tidal traffic patterns: residential and business. However, the social composition of metropolitan areas is more complex than just residential and business, and multiple social functions or services can coexist in the same location.

In [8], [9] was considered the MCN traffic from several different locations in the urban area, thus being able to capture a plurality of different types of traffic patterns. In [8], we use a real data set to study the spatio-temporal traffic fluctuations at the cell sites and at the mobile metro-core network of a Chinese City. Tidal traffic patterns were extracted as the average values of the historical data, and were used to obtain, in advance (Off-line), an optimal planning to reconfigure the mobile metro-core network at every hour of the day. The optimal planning was based on two Mixed Integer Linear Programming (MILP)s formulations for dynamic optical-resource allocation that minimize energy consumption, while providing 1+1 protection.

[2]Though Wi-Fi network traffic is not regarded as mobile data traffic, the two network scenarios are very similar.

The prediction-based optimization of [8] provide a slightly over-provisioned resources. In order to adapt to the actual real-time traffic, in [9] we proposed an On-line optimization matheuristic that takes the optimal planning results (using formulations proposed in [8]) to guide On-line routing decisions. While in [8] was considered to perform hourly reconfigurations, [9] proposes a scheduling heuristic to calculate a limited set of reconfiguration time points. The proposed scheduling provides a good trade-off between reduction of routing changes (to avoid disruption) and resource allocation efficiency (to increase energy savings).

A. Paper Contribution

In our previous work [9], both the Off-line planning and the reconfiguration time points calculation were performed once for multiple days, using average values of the historical data as the tidal traffic pattern of the network. In this paper, we introduce a machine learning-based traffic-prediction method to improve the techniques proposed in [8] and [9].

Thanks to the use of traffic prediction, the new matheuristic (Algorithm 1) reduces the complexity and required computational time of the On-line phase: in [9]) Algorithm 2 computes optimal weights in the On-line phase, while in this work such weights are computed Off-line. Moreover, we have split the Off-line phase into two phases in order to better follow the traffic changes and recompute the Off-line Planning dynamically.

The proposed method (Algorithm 1) is composed by three phases:

- *Off-line Scheduling*: predict the traffic and schedule the reconfiguration time points for the next 24 hours (see subsection IV-A).
- *Off-line Planning*: predict the traffic of the next reconfiguration interval, solve the resource allocation optimization problem for the maximum value of traffic in such interval, and calculate the optimal weights for each predicted demand (see subsection IV-B).
- *On-line Routing*: build optimal-weighted graphs (using results of Off-line planning) to compute On-line routing decision with greedy algorithms (see subsection IV-C).

While in [8], [9] we used a dataset from a middle size city of China through a collaboration with a Chinese partner. In the current work we use a public dataset from an European city, more specifically the city of Milan, Italy (presented in section VI).

III. Primer on Traffic Prediction

Before presenting the proposed matheuristic approach, in this section we provide a short introduction to traffic prediction models. Then in section V we will describe the machine learning-based model that we have proposed in this work.

Traffic prediction is a core process for network optimization decisions and a fundamental branch of machine learning. In the literature there are several prediction methods such as: ARIMA (Auto-Regressive Integrated Moving Average), F-ARIMA (Fractional-ARIMA) [16], [17], SVM (Support

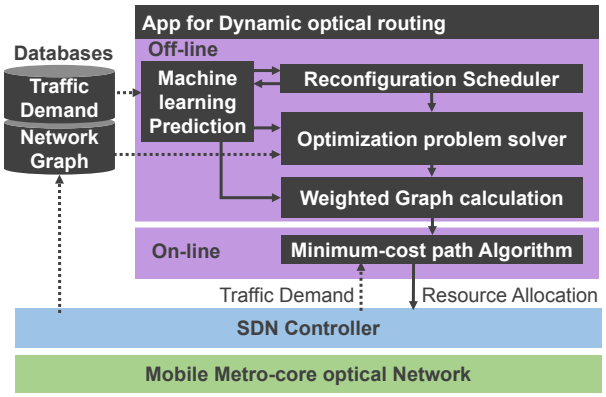

Fig. 1. Dynamic optical routing matheuristic for software-defined mobile metro-core network

Vector Machine), MLP (Multi-Layer Perceptron) and MLPWD (Multi-Layer Perceptron with Weight Decay) [18].

Performances of prediction algorithms are based on the type of data input, so we can not know if a method is better than the other until we do not try different kind of input data-sets. ARIMA models are generally applied to stationary time series with no trends or seasonality information making a regression on previous samples [19]. MLP and MLPWD are part of the Artificial Neural Network (ANN) family of algorithms. Compared to other regression methods, the ANN allows more flexible relationships and accuracy to stationary time series with no trends or seasonality information making a regression on previous samples [20].

Another good approach is to mix different methods as in [17], where authors built a hybrid model. They concluded that using ARIMA and ANN, for non-linear time series, is more efficient than using just one method. Using three different datasets (the Wolf's sunspot data [21], the Canadian lynx data [22] and the British Pound/US Dollar exchange rate data [23]), the author concluded that a mixed approach to the prediction can achieve very good results.

There are numerous approaches that have emerged through machine learning, which try to optimize routing based on traffic prediction. The authors of [24], proposed to embed a reinforcement learning module into each node of a switching network to learn topology and traffic patterns. Results of [24], show that as load increases, the learning algorithm continues to route efficiently. In [25], the authors proposed a pre-design mechanism for routing efficiency enhancement that is based in three aspects: flow feature extraction from user history data, prediction and route selection based on multi-constraint weight assignment. Instead, in [26] was proposed an on-line machine learning task that estimates the type of traffic flow and apply a very simple routing policy: elephant flows (large) are routed via a least congested path, and mouse flows (small) are routed with ECMP (Equal Cost Multiple Path) algorithm.

IV. Dynamic optical routing matheuristic

In this section, we give a description of the matheuristic depicted in Fig. 1 and described by Algorithm 1. The idea behind our proposal is to take advantage of SDN and traffic predictability to optimize On-line dynamic routing decisions.

Algorithm 1 On-line Routing Matheuristic

Phase 1 - Off-line Scheduling
It is performed on a daily basis, and it is based on the historical traffic data-set h (observation windows until the current day)
1: **Predict** the traffic demand of the next 24 hours ($\hat{h}^{24}$)
2: **Compute** $\mathcal{T}$: set of reconfiguration time points ($|\mathcal{T}| < 24$) using scheduling algorithm proposed in [9]
Phase 2 - Off-line Planning
It is performed (δ seconds) before each reconfiguration time point $t \in \mathcal{T}$. It uses historical h_d and current $\bar{h}_d^t$ traffic demands to predict traffic demand of next reconfiguration time point $\hat{h}_d^t$ and obtain the optimal resource allocation of the network in advance.
3: **for** $t \in \mathcal{T}$ **do**
At $t - \delta$ ▷ *δ seconds before the reconfiguration point*
4: **for** $d \in \mathcal{D}$ **do** ▷ *$\mathcal{D}$ Set of demands*
5: **Predict** $\hat{h}_d^t$: demand for next reconfiguration point using machine learning-based approach of section V
6: **Solve** Optimization problem using VWP or WP (subsection IV-B), for $\hat{h}_d^t$ to get optimal planning $\mathcal{S}_d^t$
7: Given optimal $\mathcal{S}_d^t$ for predicted demand $\hat{h}_d^t$
Compute $\mathcal{C}_{dr}^t$: the weights of the optimal-weighted graph for each connection request $\hat{r}_d^t = \lceil \hat{h}_d^t / L \rceil$, where L is the line rate (subsection IV-B3)
Phase 3 - On-line Routing
It is performed at each reconfiguration time point. It uses $\mathcal{C}_{dr}^t$ to guide optimal routing while running min-cost path alg.
At t ▷ *At the reconfiguration time point*
8: **for** $d \in \mathcal{D}$ **do**
Given the current traffic demand $\bar{h}_d^t$ and $\mathcal{C}_{dr}^t$
9: **for** $\bar{r}_d^t \in$ {Real-time requests belonging to $\bar{h}_d^t$ **do**
10: **Compute** routing using a greedy algorithm based on Bhandari, or on modified Bhandari [9] for VWP and WP, respectively.

In the following sections, we describe more in details the proposed machine-learning traffic prediction (Sec. V); and the datasets (Sec. VI) used to test our methodology.

A. Phase 1. Off-line scheduling

Performing hourly reconfigurations, is not well accepted by service operators because it leads to service disruption and instability of distributed routing algorithms. In section VIII we discuss how to update network rules to perform congestion free reconfiguration.

Phase 1 is done on a daily basis to reduce the reconfiguration time points ($|\mathcal{T}| < 24$) by scheduling reconfiguration events into specific time points $t \in \mathcal{T}$, creating a trade-off: a decrease of the reconfiguration time points $|\mathcal{T}|$ produce an increase of bandwidth over-provisioning (in consequence, increasing power consumption).

Off-line Scheduling phase uses the machine learning-based traffic prediction model proposed in section V to run the scheduling algorithm over a 24 hours predicted traffic demand $\hat{h}^{24}$.

Given the traffic matrix of an specific day and lower threshold of the bandwidth allocation efficiency (expected efficiency $\bar{\eta}$), The scheduling algorithm is a *Simulated Annealing*-based heuristic method that obtains a reconfiguration scheduling ($\mathcal{T}$) by finding the minimum number of reconfiguration time points ($|\mathcal{T}|$) with an expected allocation efficiency (Total Demanded Bandwidth ÷ Total Allocated Bandwidth) [9].

B. Phase 2. Off-line Planning

Phase 2 is the most complex and time consuming phase of our matheuristic. Thanks to the use of network demand forecast obtained with the machine learning-based prediction presented in section V, Phase 2 is also done an Off-line.

It is performed before each reconfiguration time point $t \in \mathcal{T}$ to obtain the optimal resource allocation $\mathcal{S}_d^t$ for each predicted traffic demand $d \in \mathcal{D}$. A short-term traffic demand prediction $\hat{h}_d^t$ of the next reconfiguration time point $t \in \mathcal{T}$ is used to:

- Solve an optimization problem to obtain the optimal resource allocation $\mathcal{S}_d^t$ for each predicted traffic demand $d \in \mathcal{D}$ during the t.
- $\mathcal{S}_d^t$ is used to calculate a set of optimal weights $\mathcal{C}_{dr}^t$ that will be use to guide the On-line routing phase decisions.

*1) **Optimization Problem**:* The Off-line planning problem consists of finding the set of optical paths that satisfies the spatio-temporal-dependent demand matrix of a specific time period t using 1+1 protection, with the objective of minimizing the energy consumption of the optical layer of the mobile metro-core network. In this work we only considered the power consumption of optical layer. The power consumption data of components are based on the models given in [27]. In the following, we introduce two MILP formulations that minimize the energy consumption of the MCN by activating and deactivating resources.

- Virtual Wavelength Path (VWP) at each OXC all wavelengths are converted to the electrical domain, allowing to perform wavelength conversion and traffic grooming to increase wavelength utilization. In VWP, the (virtual) optical path is not constrained to use the same wavelength, it can use different wavelengths on each distinct link.
- Wavelength Path (WP) at each OXC wavelengths can either be converted to electrical domain or be switched at the optical domain. WP takes advantage of the small distances in the metro-core network to establish fully transparent lightpaths with optical bypass to avoid optical-to-electrical (OE) and electrical-to-optical (EO) conversions in transit nodes[3]. However, by dropping the wavelength conversion capability, in WP an optical path is constrained to use the same wavelength on every link, the so called wavelength continuity constraint.

*2) **Pre-calculation of k pairs of link disjoint path (K-PLDP)**:* VWP and WP are multi-commodity flow problems known to be NP-complete. Therefore, we have used path formulations in order to simplify these problems and to solve them in a limited amount of time. In path formulation, instead of considering all possible paths, only a reduced set of k candidate path pairs with 1+1 protection is pre-calculated for every demand. For each demand d the k pairs of link disjoint path are calculated.

*3) **Optimal Weighted graph computation**:* In the following, we briefly describe the optimal weights computation for VWP and WP models. For WP model the On-line phase must perform routing and wavelength assignment (RWA) with a minimum cost algorithm, that can be done using a layered graph $\mathcal{G}'$.

[3]Regeneration is normally needed after 1500 km for non-coherent wavelength channels at 10 Gbit/s [27].

TABLE I
OPTIMAL LINK WEIGHTS ($c_{d\hat{r}}^{et}$) FOR VWP

Given the $\hat{r}$-th connection request of demand d at time t

Edge e condition	**Expected** $c_{d\hat{r}}^{et} \vert \hat{r}_d^t \leq \bar{r}_d^t$	**Unexpected** $c_{d\hat{r}}^{et} \vert \hat{r}_d^t > \bar{r}_d^t$
Assigned	1	Not possible
Available	$(\omega+1)$	1
Available-inactive	$(\omega+1)\Delta$	Δ
Unavailable	$\mathcal{M}$	$\mathcal{M}$

$\mathcal{M}$: *big M*, ω = Length of backup path of $\hat{r}, d, t$
Δ: *fiber-to-wavelength activation cost ratio* ($\Delta > 0$)

- *Optimal weights for VWP model* For each $\hat{r} \in \{1..\hat{r}_d^t\}$ belonging to demand d at reconfiguration point t, a set of weights $\mathcal{C}_{dr}^t$ is generated. Table I summarizes the possible link weights $c_{d\hat{r}}^{et}$ that can be assigned to the vertex-weighted graph $\hat{\mathcal{G}}$. Based on the current traffic demand $\bar{h}_d^t$ and the predicted traffic demand $\hat{h}_d^t$, there are two scenarios: *i*) *Expected* ($\hat{r}_d^t \leq \bar{r}_d^t$): all requests $\hat{r}$ are optimally planed. *ii*) *Unexpected* ($\hat{r}_d^t > \bar{r}_d^t$): a sub-set of the requests are optimally planed $\{\hat{r} | \hat{r} \leq \bar{r}_d^t\}$, while the rest are unexpected $\{\hat{r} | \hat{r} > \bar{r}_d^t\}$. Based on the Off-line planning results, for each request $\hat{r} \in \{1..\hat{r}_d^t\}$ there are 4 possible link $e \in \mathcal{E}$ conditions:
  - *Assigned:* e belongs to the $\hat{r}$-th pair of paths.
  - *Available:* e has at least 1 free wavelength in active fibers.
  - *Available-inactive:* e has at least 1 inactive fiber.
  - *Unavailable:* e has no free wavelengths.
- *Optimal weights for WP model*
  In WP model wavelengths need to be assigned to the lightpaths. Therefore, we use the equivalent weighted-layered-graph (see Figure 2) transformation $\hat{\mathcal{G}}'$ proposed in [28] to reduce the RWA to a minimum-cost path algorithm. We extend this method the RWA with 1+1 protection by finding a pair of edge-disjoint paths in $\hat{\mathcal{G}}'$. Given the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ (defined in section IV-B), its equivalent layered-graph representation is: $\mathcal{G}'(\mathcal{V}', \mathcal{E}', \mathcal{V}^s, \mathcal{A}^s, \mathcal{V}^d, \mathcal{A}^d)$, where:
  - Each node and link from $\mathcal{G}$ are replicated WF times as virtual nodes $v^{\lambda f} \in \mathcal{V}'$ and links $e^{\lambda f} \in \mathcal{E}'$; for each wavelength $\lambda \in \{1..W\}$ on every fiber $f \in \{1..F\}$.
  - For each node in $\mathcal{G}$ two dummy nodes are added to map source and destination of the demands $d \in \mathcal{D}$. Dummy source node $v^s \in \mathcal{V}^s$ related to node $v \in \mathcal{V}$ has only outgoing dummy-arcs $\mathcal{A}^s$ towards the virtual replicas of v ($v^{\lambda f}$). Dummy destination node $v^d \in \mathcal{V}^d$, has only incoming dummy-arcs $\mathcal{A}^d$ from the $v^{\lambda f}$.

  For each $\hat{r} \in \{1..\hat{r}_d^t\}$ belonging to demand d at the next reconfiguration point t, a set of weights ${\mathcal{C}'}_d^t$ is generated. Table II summarizes the 10 possible weights ${c'}_{d\hat{r}}^{e't}$ that can be assigned to virtual links ($e^{\lambda f}$, hereafter e' for simplicity) of the vertex-weighted layered-graph $\hat{\mathcal{G}}'$. There are two type of requests: expected and unexpected

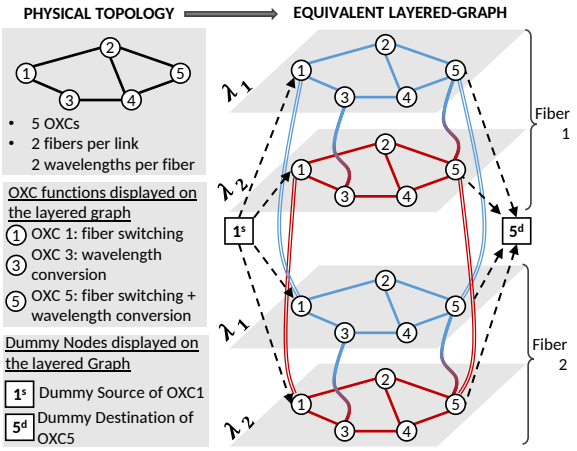

Fig. 2. Equivalent layered-graph $\mathcal{G}'$ for the physical topology shown in the top left corner, and considering only two dummy nodes $1^s \in \mathcal{V}^s$ and $5^d \in \mathcal{V}^d$ where a demand from OXC1 to OXC5 can be mapped.

TABLE II
OPTIMAL VIRTUAL LINK WEIGHTS ($c^{e't}_{d\hat{r}}$) FOR WP

Given the $\hat{r}$-th connection request of demand d at time t

Virtual Link $e'(e^{\lambda f})$ **condition**	**Expected** $c^{e't}_{d\hat{r}} \mid \hat{r}^t_d \leq \bar{r}^t_d$	**Unexpected** $c^{e't}_{d\hat{r}} \mid \hat{r}^t_d > \bar{r}^t_d$
Working	σ	Not possible
Backup	1	Not possible
Available	$(\omega+1)$	1
Inactive	$(\omega+1)\Delta$	Δ
Unavailable	$\mathcal{M}$	$\mathcal{M}$

$\mathcal{M}$: *big M*, $\sigma << 0$, ω = Length of backup path of $\hat{r}^t_d$
Δ: *fiber-to-wavelength activation cost ratio* ($\Delta > 0$)

demands. Based on Off-line optimal planning S_t, there are 5 possible conditions of links $e' \in \mathcal{E}'$:

- *Working:* e' belongs to the $\hat{r}$-th working path of d.
- *Backup:* e' belongs to the $\hat{r}$-th backup path of demand d.
- *Available:* e' is a free wavelength of an active fiber.
- *Inactive:* e' belongs to an inactive fiber of e.
- *Unavailable:* e' is assigned to other request or demand.

In the layered graph $\hat{\mathcal{G}}'$, the cost of dummy arcs is always equal to zero $c_a = 0$.

C. Phase 3. On-line Routing

Not by chance, phase 3 is the fastest phase of Algorithm 1, allowing to take routing decisions On-line[4].

On-line routing phase is based on a heuristic method that favors optimality by running a minimum cost algorithm on a set of optimally vertex-weighted graphs $\hat{\mathcal{G}}^t_{dr}$ that are built with $\mathcal{C}^t_{dr}$. The minimum-cost path algorithm computes a pair of link-disjoint paths for each request $\hat{r} \in \{1..\hat{r}^t_d\}$ of each demand $d \in \mathcal{D}$ at reconfiguration time point $t \in \mathcal{T}$.

For VWP we use Bhandari's pair of link disjoint paths algorithm [29]. In WP, due to the wavelength continuity constraint, we need to perform routing and wavelength assignment (RWA) with a minimum-cost algorithm. This is possible using a layered graph $\mathcal{G}'$ (see Figure 2), where each link of the original graph $\mathcal{G}$ is replicated WF times. However, virtual-edge disjointness in $\mathcal{G}'$ do not guarantee physical edge disjointness in $\mathcal{G}$. Thus, in WP it is used a modification of Bhandari's link-disjoint path algorithm that can be applied to the layered graph [9].

[4]In our results the On-line Routing phase obtains the routing decisions in less than 60 μs

V. MACHINE LEARNING-BASED TRAFFIC PREDICTION

Our approach is to use forecasted traffic load to calculate in advance the best resource allocation in the optical metro network to reduce its energy consumption. To obtain the traffic forecast we deployed a machine learning-based model.

In machine learning, an ANN is a network inspired by the central nervous systems of animals, which are used to estimate or approximate functions that can depend on a large number of inputs that are generally unknown [20]. We model a feed forward neural network with the aim to forecast from one to 24 hours of traffic demands for each base station (or aggregation ring node, see Fig. 4) of the mobile network operator.

Each time we describe a neural network algorithm we will typically specify the *Architecture*, the *Activity rule* and the *Learning rule*.

- *Architecture*: the architecture specifies which variables are involved in the network and their topological relationships. For example, the variables involved in a neural network might be the number of layers, neurons, weights of the connections between the neurons. Our model is composed by two layers: one hidden layer with five nodes, and one output layer with one node, see Fig.3. The input is composed by six entries:
  - $i_{s,1}$ Hour of the day;
  - $i_{s,2}$ Day of the week;
  - $i_{s,3}$ A flag for holiday/weekend;
  - $i_{s,4}$ Previous day's average load;
  - $i_{s,5}$ Load from the same hour of the previous day;
  - $i_{s,6}$ Load from the same hour and same day from the previous week.

  The result of the output node $y_{s,1}$ represents the hour that we want to predict.
- *Activity rule*: most neural network models have local rules and define how the activities of the neurons change in response to each other. We define the activation function of the neuron as the sigmoid function, Equation 1, useful for regression problems:

$$f(x) = \frac{1}{1+e^{-x}} \tag{1}$$

where x is the sum of the input weights multiplied by the output value of the node from the previous layer. Equation 2 represents the vectorial representation of the forwarding procedure; Instead, Equation 3 represents its analytical form:

$$Y = I \times W^{(1)} \times W^{(2)} \tag{2}$$

$$y_{s,z} = f\left[\sum_{h=1}^{H} w_{h,z} \cdot f\left(\sum_{n=1}^{N} w_{n,h} \cdot i_{s,n}\right)\right] \quad \forall s \in [1,S] \tag{3}$$

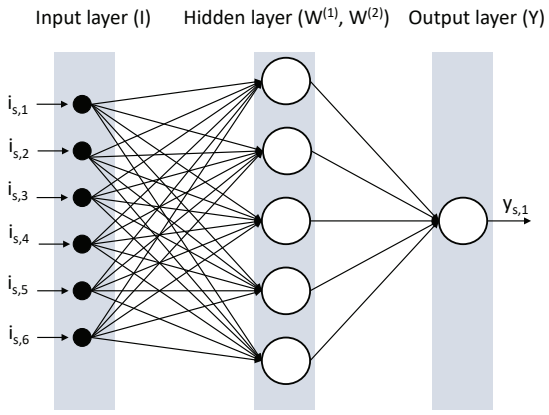

Fig. 3. Architecture of the Artificial Neural Network.

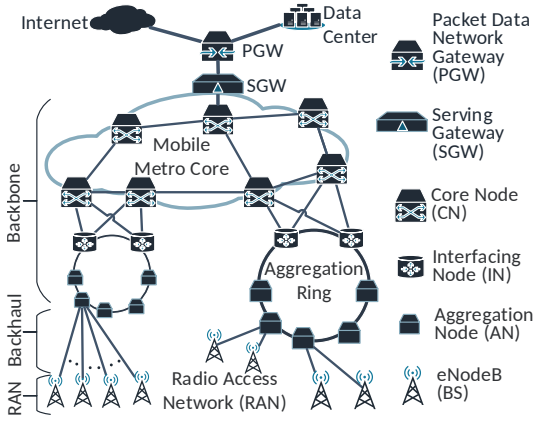

Fig. 4. Reference mobile carrier network (MCN) architecture.

where:

- S, number of samples
- $N = 6$, number of inputs
- $H = 5$, number of hidden nodes
- $Z = 1$, number of output nodes
- $I \in \mathbb{R}^{[S \times N]}$, each element represents the single input and it is in the form $i_{s,n}$, where $s \in [1, S]$ and $n \in [1, N]$.
- $W^{(1)} \in \mathbb{R}^{[N \times H]}$, each element represents the weight from input node n to hidden node h, and it is in the form $w_{n,h}$, where $n \in [1, N]$ and $h \in [1, H]$.
- $W^{(2)} \in \mathbb{R}^{[H \times Z]}$, each element represents the weight from hidden node h to the output node z, and it is in the form $w_{h,z}$, where $h \in [1, H]$ and $z \in [1, Z]$.
- $Y \in \mathbb{R}^{[S \times Z]}$, each element represents the single output and it is in the form $y_{s,z}$, where $s \in [1, S]$ and $z \in [1, Z]$.

- *Learning rule*: the learning rule specifies the way in which the neural network's weights $\left(W^{(1)} \text{ and } W^{(2)}\right)$ change with time. Typically a learning rule is an objective function that measure how well the network with weights solves the task. The training process adjusts the weights to minimize the objective function, using a form of gradient descent algorithm called Levenberg-Marquardt [30].

Once trained the model, as showed in section VII-B, we performed the prediction of the internet traffic and used the results for the optimization phase.

VI. The Milan mobile carrier network datasets

In a software-defined mobile metro-core network (SD-MCN), the topology and current state of the network will be provided by the SDN controller (typically through a REST interface) to the application layer network database, as depicted in Fig. 1. In the same way, the traffic demand is provided by the SDN controller to the application layer in order to: *i*) perform On-line routing and *ii*) update the traffic demand database that is used by the learning process of the prediction algorithm.

However, in this work we did not have access to a real and deployed MCN, therefore we collected such databases manually as described in the following subsections.

A. MCN Datasets

The MCN data used in this work is composed by the following two datasets [31].

1) The first one refers to the traffic of voice/sms/data of Milan city, measured during November and December 2013 [32]. It is the result of a computation over the Call Detail Records (CDRs) generated by the Telecom Italia cellular network. CDRs log the user activity for billing purposes and network management every ten minutes, creating 144 records for each day. The data-set contains the following information: Square cell ID (from 1 to 10000), Time interval, Country code, Received SMS, Sent SMS, Received Calls, Sent Calls, Internet.
   We have assumed that the capacity of the MCN cells is 1 Gbps according to recent 4G commercial solutions [5].
2) The second dataset provides the location of 2554 base stations of TIM (Italy's incumbent communication service provider) deployed in Milan [34], such as: Base Station ID (from 1 to 2554), Latitude and Longitude.

B. MCN Architecture

In this paper we follow the MCN architecture described in [8], [9], which is based on LTE current deployment. However, the same architecture remains valid to support a future evolution to 5G.

1) The MCN architecture elements: As depicted in Fig. 4, the MCN is commonly modelled with a three-level hierarchical architecture, composed by the radio access, the *back-haul* and the *backbone* networks.

The *base stations* (BS)s deployed on the field provide radio access. Each BS (comprising a set of cell antennas and an eNodeB) is connected by a back-haul network segment to an *aggregation node* (AN). The ANs are the edge elements interfacing the back-haul to the backbone network. The backbone network is the infrastructure connecting the ANs to the *serving gateway* (SGW).

We assumed that the metro backbone is divided into an aggregation and a metro-core segment, in order to gradually groom traffic of the metro area from the edges towards the

[5]Since the beginning of 2016 commercial 4G devices support downlink data speeds of 1 Gbps [33].

SGW. The *aggregation network* is composed by metro optical rings, each one connecting a subset of neighbor ANs. On every ring, two ring nodes, called *interfacing nodes* (IN)s, are used to interface the aggregation to the core segment of the backbone network. Each IN is connected to a *core node* (CN) of the core network (Fig. 4). The *mobile metro-core network* is the mesh-topology fiber infrastructure interconnecting the core nodes and the SGW. The assumption of mesh topology in the metro-core is consistent with the current evolutionary trend leading from ring to mesh in metro areas to reduce latency and increase reliability [35], [36]. We suppose that each node of the core network is an optical cross connect (OXC).

The SGW is connected to a *packet gateway* (PGW) which provides connectivity towards data center facilities and Internet Exchange Points (IXP)s. It is important to notice that all the mobile data traffic must pass through the metro SGW: therefore the part of network which extends from the SGW to the PWG and beyond has not been included in this study.

2) The MCN Protection: Given that the connections in the MCN transports large volumes of traffic to/from aggregation rings, and should meet service level agreements to offer carrier grade services, we assumed that these connections need to be provisioned with 1+1 protection scheme [37]. The combination of aggregation rings and mesh core, together with the dual-homed interconnection of each ring to the core, enables full resilience of the physical infrastructure of the entire mobile metro backbone against (at least single) failures. The methodologies proposed in section IV introduce 1+1 protection with a pair of edge disjoint paths, which establishes the active and backup path on different INs of each ring. Offering 1+1 protection is still simple, because it can be seen as the establishment of two link-disjoint dedicated connections for each request instead of a single path. Moreover, we avoided the shared path protection scheme to keep the problem simpler [38].

The MCN topology was synthesized from the real geographical location of 2554 BSs in Milan city [34], using the following methodology.

3) Methodology to synthesize the MCN topology:

- RAN: based on BSs location, a clustering algorithm creates groups of up to 15 BSs with minimum distance between BSs and AN of the cluster.
- Aggregation rings: based on ANs, a second level of clustering creates 8 aggregation rings of up to 20 ANs each. Each aggregation ring has two interfacing nodes (INs) towards the metro-core.
- Metro-core: it is composed by 16 ($8x2$) CNs and one SGW, connected by multi-fiber links (with 80 wavelengths per fiber) in a maximal planar graph with degree 6. Finally, we perform network dimensioning by solving a multi-commodity problem to minimize CapEx, assuming that every ring is generating traffic at its daily peak. The multi-commodity flow problem must be adapted to each model introduced in section IV-B, however to simplify the presentation, we do not describe them. The resulting network for WP model is shown in Fig. 5.

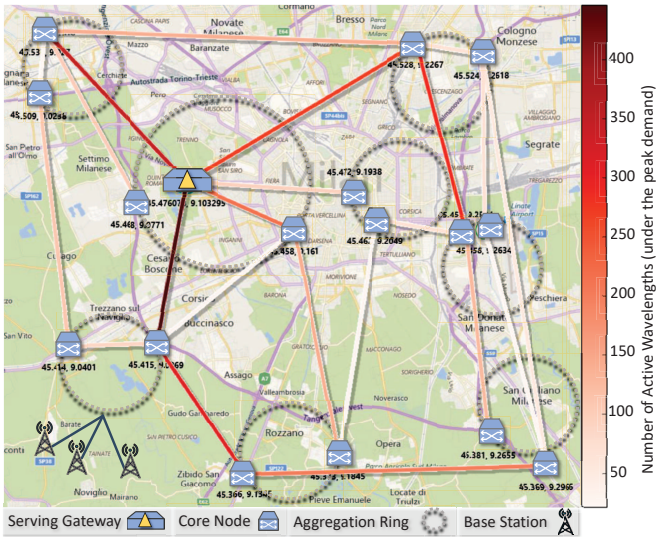

Fig. 5. MCN topology of Milan city, synthesized with methodology reported in subsection VI-B3 using geo-location of 2554 BSs. Serving gateway placed at Milan Internet Exchange (MIX). Number of active wavelengths dimensioned for WP model. Aggregation rings (composed by up to 20 aggregation nodes and two interfacing nodes) and BSs are shown for illustrative purpose.

VII. Results

In this section we present a complexity and time consumption analysis of the proposed matheuristic (Algorithm 1), the performance analysis of the prediction model presented in section V, and the power consumption of the mobile metro-core network presented in [8] using several methods for the datasets presented in section VI. The results are exposed two specific days: Friday 16 and Saturday 17 of December, 2013.

A. Computational Complexity and Time Assessment

Table III summarizes the complexity of each phase of Algorithm 1 in terms of variables (and constraints for MILP formulations). Moreover, Table III provides the computational time required by each phase of the proposed approach, when using a machine with an *IntelCore i7-6700HQ at 2.6 GHz* processor with *16384 MB DDR4 at 1066.1 MHz* of memory.

1) Phase 1. Off-line Scheduling: it is performed once per day, and it is dominated by the ANN traffic prediction algorithm.

2) Phase 2. Off-line Planning: it is performed at each reconfiguration time point and it is dominated by the computation of optimal weights. As expected, in Table III it is clear that WP requires more time than VWP approach, due to the wavelength continuity constraint enforcement.

3) Phase 3. On-line Routing: it is performed for each traffic demand request that arrives to the network and it is dominated by the computation of routing paths. This phase is very fast (less than 60 μsec in average) as it only involves solving greedy algorithms for each demand. As expected, WP model consumes more time as it is solved in a layered graph.

Regarding the prediction algorithm, one of the great advantages of the neural networks is the time computation. The ANN algorithm performs a limited number of operation during the training phase. For each epoch it computes the gradients to derive the weights, then the complexity is proportional to

TABLE III
COMPLEXITY AND TIME COMPUTATION ASSESSMENT

Phase	Complexity		Computational Time VWP	Computational Time WP
Phase1. Off-line Scheduling	ANN Prediction (next 24 hours)	$O(TVH(N+Z))$	$8.69s$	
	Scheduling	$O(Vm^2)$		
Phase 2. Off-line Planning	ANN - Prediction (next reconfiguration interval)	$O(TVH(N+Z))$		
	MILP problems	VWP Constraints: $O(V(V-1)R)$ VWP Variables: $O(V(V-1)RK)$ WP Constraints: $O(V(V-1)R(K+1))$ WP Variables: $O(V(V-1)RKW)$	$13.09s$	$555,5s$
	Computation of optimal weights	VWP: $O(V^2(V-1)^2RE)$ WP: growth with $O(V^2(V-1)^2REWF)$		
Phase 3. On-line Routing	Build optimal-weighted graph	VWP: $O(E)$ WP: $O(WFE)$	$50\mu s$	$60\mu s$
	Computation of routing paths	VWP: Bhandari $O(K(V+ElogE))$ WP: modified Bhandari $O(KWF(V+ElogE))$		

m maximum number of reconfiguration time points. T, N, H, Z number of epochs, input, hidden and output nodes of ANN. V, E number of aggregation rings and directed links. W wavelengths per fiber. F maximum number of fibers per link. R and K the average number of connection requests per each demand and the number of pairs of link disjoint paths.

the number of weights to update. Since our architecture is composed just by one hidden layer with 5 nodes, the algorithm takes 0.966 seconds in medium for each metro node. In total, counting the nodes, the algorithm takes 7.72 seconds for the metro optical network.

It is important to notice that thanks to the use of traffic prediction, the most time consuming tasks of Algorithm 1 are performed Off-line. Thus the introduced overhead required to perform our optimization method corresponds only to the On-line phase, which is less than $60\mu s$, allowing to be implemented even in scenarios with very stringent requirements of delay, such as 5G [35], [36].

B. Traffic prediction

Before making the prediction we have to train the model for each node of the metro network, this means that we have a neural network for each node. First of all we divided the dataset described in section VI in three part: training, validation and test set. The training set goes from November 1th to December 10th; the validation set goes from December 11th to December 15th; and the test set from December 16th to December 18th

The validation is a small part of the training set used to validate the training, by which we can avoid over-fitting problems.

1) Training: We used the training dataset as input for the neural network in order to derive the weight matrices $W^{(1)}$ and $W^{(2)}$ (Fig. 3). In particular, we used a form of Gradient Descent called Levenberg-Marquardt [30] that minimize the error between the actual value and the forecasted one, performing the training in a supervised way. In order to minimize the error and to avoid the problem of over-fitting, we trained the model for a number of times (epochs) that goes from 1 to 1000. The over-fitting is a typical problem in the prediction models and it occurs when a model begins to memorize training data rather than learning to generalize from trend. Accordingly, we kept trace of the prediction error at each epoch applying the trained model to the validation set, and stopped the training when the error started to grow. Once trained the model, we stored the weight matrices and test the model with the test set.

2) Test: We made the test of the model by using the test set as input and deriving the output thanks to the feed-forward formula in the Equation 3. The traffic prediction performance have been evaluated by obtaining three parameters: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and the Root Mean Square Error (RMSE) overall the network. As shown in Table IV, the percentage error at the aggregation level is smaller than the one at the base station level. The last happens because at the base station level the traffic is directly influenced by many factors such as: type of the day (weekday or weekend), special events, users mobility pattern, etc., for this reason many peaks could not be predicted accurately. After aggregating the traffic, it becomes more stationary and regular, as showed in Fig. 6, indeed the neural network model gives better results.

C. Optical routing

In order to assess the performance of the optical routing techniques, a Discrete event simulator (DES) was built using SimPy, a process-based DES framework based on Python. In this section we compare the performance of several optical routing techniques in terms of total power consumed (kW) by

TABLE IV
TRAFFIC PREDICTION PERFORMANCE

Aggregation Ring			Base Station		
MAE	MAPE	RMSE	MAE	MAPE	RMSE
$2.52{\cdot}10^3$	3.81%	$3.23{\cdot}10^3$	12.26	9.2 %	15.82

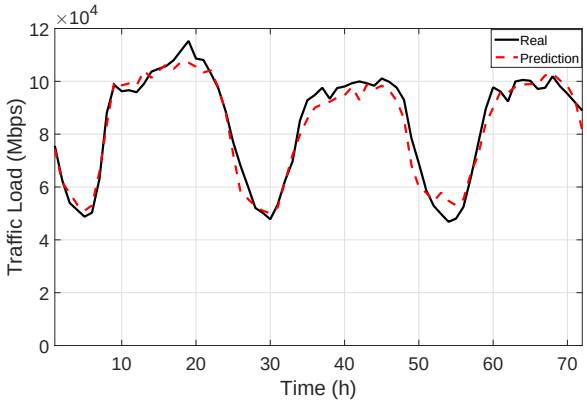

Fig. 6. Comparison between the real traffic load and the machine learning based prediction, for a core node of the synthetic topology (Fig. 5), during 16^{th}, 17^{th} and 18^{th} of December 2013.

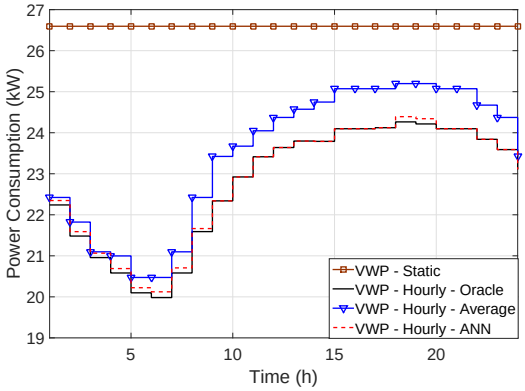

Fig. 7. Optical layer power consumption of the mobile metro-core network in Milan city during 16/12/2013, using four different optical resource allocation techniques based on VWP

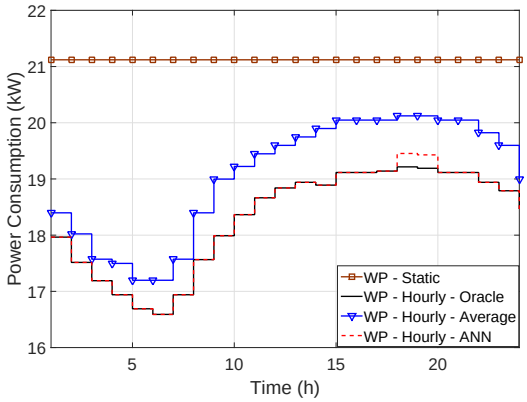

Fig. 8. Optical layer power consumption of the mobile metro-core network in Milan city during 16/12/2013, using the six different optical resource allocation techniques with WP

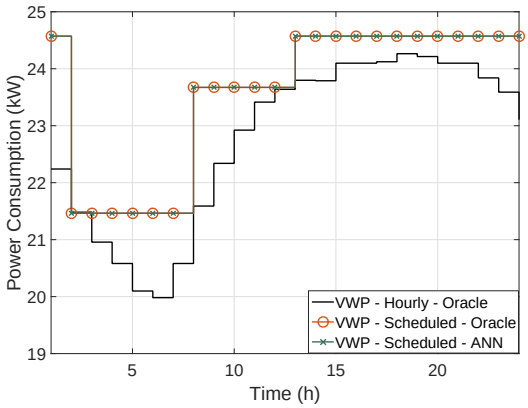

Fig. 9. Optical layer power consumption of the mobile metro-core network in Milan city during 16/12/2013, using: VWP-Hourly-Oracle, VWP-Scheduled-Oracle and VWP-Scheduled-ANN

the optical layer of the synthetic MCN of Milan city. Fig. 5 shows the topology used for WP-based techniques (number of wavelengths per link were dimensioned for WP model). When dimensioning the topology for VWP model, the number of wavelengths per link varies, however the VWP topology was omitted to simplify the presentation of this work.

By modifying or shutting down some building blocks of the matheuristic proposed in section IV, we can define a set of different optical routing techniques. In the following we describe several techniques, that can be applied either using WP or VWP models:

- *Static*: this is the current method of operation, where all the elements are active to cope with the peak hour demand of the historical data set.
- *Hourly-Oracle*: hourly reconfigurations, Off-line planning based on the solution of optimization models for a perfect traffic prediction (oracle).
- *Hourly-Average*: hourly reconfigurations, Off-line planning based on the solution of optimization models for the average traffic pattern, and On-line routing based on matheuristic reported in [9]. In fact, *Hourly-Average* corresponds to the case of *On-line opt. weight* in [9].
- *Hourly-ANN*: hourly reconfigurations, Off-line planning based on the solution of optimization models for the ANN-based traffic prediction (section V) using one hour horizon, and On-line routing based on the matheuristic reported in [9].
- *Scheduled-Oracle*: scheduled reconfigurations, Off-line planning using a perfect traffic prediction, and On-line routing using minimum-cost algorithm.
- *Scheduled-ANN*: this is the proposed Algorithm 1: Off-line scheduling using 24 hour ANN-based traffic prediction, Off-line planning using ANN-based prediction of the next reconfiguration time point, and On-line routing using minimum-cost algorithm.

Fig. 7 and Fig. 8 depicts the energy consumption when applying static network configuration and three different techniques with hourly network reconfigurations (oracle, average and ANN), using VWP and WP, respectively. In both Figures, the major energy dissipation E (kWh) comes from the flat lines that represent the over-provisioned static network configuration. WP-static (Fig. 8) consumes 20.5% less energy than VWP-static (Fig. 7), thanks to the use of optical bypass. The rest of the curves with a step-wise behavior, correspond to minimization of active resources hourly ($|\mathcal{T}| = 24$). Table V summarizes the energy dissipation and the optimality gap of dynamic routing techniques with hourly reconfigurations.

During the first experiment day we note that the VWP-Hourly-Average and WP-Hourly-Average curves allows to reduce E by 11.6% and 9.7%, respectively when compared with the static case. However the use of average traffic pattern of previous days cannot follow the daily variations of the traffic leading to an optimality gap of 3% (VWP-Hourly-Average) and 4% (WP-Hourly-Average), when compared to the oracle benchmarks [6]: VWP-Hourly-Oracle and WP-Hourly-Oracle, respectively.

Figures 7 and 8 demonstrate the effectiveness of the proposed matheuristic with ANN prediction, X-Hourly-ANN curves depict an almost perfect match with the X-hourly-Oracle. The use of traffic prediction allows reduce the optimality gap below 0.2% (VWP-Hourly-ANN) and 0.45% (WP-Hourly-ANN), while the matheuristic with average traffic pattern displays optimality gap above 3%, and a simple heuristic based on fixed weights to perform On-line routing (no Off-line planning) reports an optimality gap of almost 10% [9]. In the second part of the Table V we showed even the performances of the second experiment day.

The curves with long steps in Fig. 9 represent the results of applying the Off-line scheduling phase as proposed by algorithm 1. For comparison purpose Fig. 9 shows also the VWP-Scheduled-Oracle, and the VWP-Hourly-Oracle. As in the hourly results, the ANN prediction allows to do On-line routing that performs as good as the the Off-line planning (based on perfect prediction) with reconfiguration. Here, we have shown how to optimize network resources only in certain times of the day thanks to the use of a variable predictor based on neural networks. This promising result is part of a preliminary work that will guide us in the optimization of strategic reconfiguration points and significantly reduce the service disruption rate, exploiting also new ways to optimize the network.

VIII. Open Issues

In this section we provide a discussion on two open issues that can be considered to improve our methodology.

1) Network rules update: While Algorithm 1 uses an Off-line-scheduling phase to reduce the number of reconfigurations per day to perform, it only computes the new network state. However, Algorithm 1 does not specify the set of actions to enforce a quick and congestion-free reconfiguration of the network.

For instance, [39] presented *SWAN*, a system that reconfigures the network's data plane, based on current traffic demand, in a congestion-free mode to maximize network utilization. The key points of this system are the following: 1) reservation of a scratch capacity s (e.g., 10%) for each link in order to avoid congestion situation; 2) an algorithm that finds a congestion free plan with a minimum number of steps that is at most $\lceil 1/s \rceil - 1$. Testbed experiments and data-driven simulations show that SWAN can carry 60% more traffic than the current practice (e.g., MPLS-TE) when applying reconfigurations every 5 minutes with real-time traffic flows. From time point of view, computes allocation and rule plan in $1.3s$, congestion-controlled plan in $0.7s$ and change openflow switch rules in $0.6s$. Other systems, such as *Dionysus* [40] and *zUpdate* in [41], have explored smart ways to re-configure the network routing rules at the nodes as well. *Dionysus* achieves fast, consistent network updates through dynamic scheduling of rule updates. Instead of statically selecting an order (as in *SWAN*), this method implements on-the-fly ordering based on the real-time behavior of the network and the switches. This approach allows to improve the median network update speed by 53%-88% over static scheduling.

In order to avoid extra delay and overhead to the dynamic routing, we could add another Off-line phase that implements a reconfiguration system to minimize the delay due to the change of routes, and avoids congestion of links.

2) Cross-layer Optimization: While SDN allows to gather multi-layer and multi-domain control and visibility, we only focused on the optical layer. A future work might include a cross-layer optimization to take into account multiple layers, such as IP and optical.

The WP model has little impact on the IP layer, it exploits the relatively small distances of metro networks to establish all-optical connections, reducing the flows that need to be processed at the IP layer, and in consequence reduces delay, energy and costs in the network [27]. VWP make extensible use of the IP layer, as it uses traffic grooming at every node, with the cost of adding more load to the IP layer of the network, but increasing the wavelengths utilization. An advanced traffic grooming approach can be explored to establish a smart selection of all-optical paths and traffic grooming points. Such approach can optimize the aggregation of IP flows into the optical layer [15].

IX. Conclusions

This paper proposes an effective matheuristic with ANN traffic prediction for energy efficient dynamic optical routing in mobile metro-core networks. Exploiting the programmability and full network visibility leveraged by SDN in the mobile metro-core network, this technique can be deployed as an SDN application to perform reconfiguration of the network based on historical and current traffic-load.

Our models were tested by synthesizing a network topology from real cell cite locations, and a real traffic dataset from the city of Milan, Italy. Our results demonstrate that the use of traffic prediction represents an essential component to optimize the network with dynamic optical routing or other advanced techniques. Moreover, the mobile metro-core networks provide highly predictable aggregated traffic patterns, which was proven to be a valuable feature for network optimization. Another feature of the mobile metro-core network is the relatively small link length, that allows to exploit optical bypass capabilities to avoid costly optical-to-electrical and electrical-to-optical conversions.

The proposed matheuristic with ANN prediction, that performs On-line optical routing, reported energy consumption levels of a network that is configured by solving an optimization problem with a perfect traffic prediction (Oracle). When

[6]VWP-Hourly-Oracle and WP-Hourly-Oracle are considered as benchmarks because they represent a system with perfect traffic prediction (oracle), therefore the energy consumption is the same as solving the Optimization models and gives the best results we can get.

TABLE V
ENERGY DISSIPATION AND OPTIMALITY GAP, IN MILAN CITY DURING 16-17 DECEMBER OF 2013

	VWP				WP			
16/12/2013	Static	Hourly (Oracle)	Hourly (Average)	Hourly (ANN)	Static	Hourly (Oracle)	Hourly (Average)	Hourly (ANN)
Total Energy (kWh)	638.28	546.93	563.86	548.08	506.88	439.28	457.66	439.76
Energy Saving (compared to static)		14.3%	11.6%	14.1%		13.3%	9.7%	13.2%
Optimality Gap (with oracle)			3%	0.2%			4%	0.1%
17/12/2013	Static	Hourly (Oracle)	Hourly (Average)	Hourly (ANN)	Static	Hourly (Oracle)	Hourly (Average)	Hourly (ANN)
Total Energy (kWh)	638.28	548.90	563.86	549.16	506.88	442.62	457.66	444.64
Energy Saving (compared to static)		14%	11.6%	13.9%		12.6%	9.7%	12.2%
Optimality Gap (with oracle)			2.6%	0.04%			3.2%	0.45%

comparing VWP-Static, the common approach in today's mobile metro-core networks, with WP-Hourly-ANN energy savings of 31% can be achieved, thanks to load adaptive network operation and optical bypass. Finally, all the advantages reported by our methods can be achieved with a time overhead of less than 60 μs, thanks to a very lightweight On-line phase, and more complex Off-line phases.

REFERENCES

[1] Cisco Visual Networking Index, "Global mobile data traffic forecast update, 2015-2020," 2016. [Online]. Available: http://www.cisco.com/

[2] (2017) Nolle: In 2017, cost per bit exceeds revenues. [Online]. Available: http://www.lightreading.com/

[3] D. Kreutz *et al.*, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, Jan 2015.

[4] R. Mijumbi *et al.*, "Network function virtualization: State-of-the-art and research challenges," *IEEE COMST*, vol. 18, no. 1, pp. 236–262, Firstquarter 2016.

[5] C. Song *et al.*, "Limits of Predictability in Human Mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.

[6] Z. Zhong *et al.*, "Considerations of effective tidal traffic dispatching in software-defined metro ip over optical networks," in *OECC*, June 2015, pp. 1–3.

[7] Z. Niu, "Tango: traffic-aware network planning and green operation," *IEEE Wireless Commun.*, vol. 18, no. 5, pp. 25–29, October 2011.

[8] R. Alvizu *et al.*, "Energy aware optimization of mobile metro-core network under predictable aggregated traffic patterns," in *IEEE ICC*, May 2016, pp. 1–7.

[9] R. Alvizu *et al.*, "Energy efficient dynamic optical routing for mobile metro-core networks under tidal traffic patterns," *JLT*, vol. 35, no. 2, pp. 325–333, 2017.

[10] M. Afanasyev *et al.*, "Analysis of a Mixed-use Urban Wifi Network: When Metropolitan Becomes Neapolitan," in *ACM SIGCOMM Conf. Internet Measurement*, 2008, pp. 85–98.

[11] C. Lange and A. Gladisch, "Energy Efficiency Limits of Load Adaptive Networks," in *OFC*, 2010, p. OWY2.

[12] C. Peng *et al.*, "Traffic-driven Power Saving in Operational 3G Cellular Networks," in *Int. Conf. Mobile Computing Netwo.*, 2011, pp. 121–132.

[13] L. Budzisz *et al.*, "Dynamic Resource Provisioning for Energy Efficiency in Wireless Access Networks: A Survey and an Outlook," *IEEE COMST*, vol. 16, no. 4, pp. 2259–2285, Oct-Dec 2014.

[14] R. Wang *et al.*, "Energy saving via dynamic wavelength sharing in twdm-pon," *IEEE JSAC*, vol. 32, no. 8, pp. 1566–1574, Aug 2014.

[15] Z. Zhong *et al.*, "Energy efficiency and blocking reduction for tidal traffic via stateful grooming in ip-over-optical networks," *JOCN*, vol. 8, no. 3, pp. 175–189, Mar 2016.

[16] R. Li *et al.*, "The learning and prediction of application-level traffic data in cellular networks," *arXiv preprint arXiv:1606.04778*, 2016.

[17] G. Zhang, "Time series forecasting using a hybrid arima and neural network model," *Neurocomputing*, vol. 50, pp. 159 – 175, 2003.

[18] A. Y. Nikravesh *et al.*, "Mobile network traffic prediction using mlp, mlpwd, and svm," in *IEEE BigData Congress*, 2016, pp. 402–409.

[19] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. OTexts, 2014.

[20] D. C. Park *et al.*, "Electric load forecasting using an artificial neural network," *IEEE Tran. Power Sys.*, vol. 6, no. 2, pp. 442–449, May 1991.

[21] K. W. Hipel and A. I. McLeod, *Time series modelling of water resources and environmental systems*. Elsevier, 1994, vol. 45.

[22] M. Campbell and A. Walker, "A survey of statistical work on the mackenzie river series of annual canadian lynx trappings for the years 1821-1934 and a new analysis," *Journal of the Royal Statistical Society. Series A (general)*, pp. 411–431, 1977.

[23] R. A. Meese and K. Rogoff, "Empirical exchange rate models of the seventies: Do they fit out of sample?" *Journal of international economics*, vol. 14, no. 1-2, pp. 3–24, 1983.

[24] J. A. Boyan, M. L. Littman *et al.*, "Packet routing in dynamically changing networks: A reinforcement learning approach," *Advances in neural information processing systems*, pp. 671–671, 1994.

[25] F. Chen and X. Zheng, "Machine-learning based routing pre-plan for sdn," in *Int. Workshop on Multi-disciplinary Trends in AI*, 2015, pp. 149–159.

[26] P. Poupart *et al.*, "Online flow size prediction for improved network routing," in *IEEE ICNP*, 2016, pp. 1–6.

[27] W. Van Heddeghem *et al.*, "Power consumption modeling in optical multilayer networks," *Photonic Network Commun.*, vol. 24, no. 2, pp. 86–102, 2012.

[28] C. Chen and S. Banerjee, "A new model for optimal routing and wavelength assignment in wavelength division multiplexed optical networks," in *IEEE INFOCOM*, vol. 1, Mar 1996, pp. 164–171 vol.1.

[29] R. Bhandari, *Survivable Networks: Algorithms for Diverse Routing*. Norwell, MA, USA: Kluwer Academic Publishers, 1998.

[30] H. B. Demuth *et al.*, *Neural network design*. Martin Hagan, 2014.

[31] S. Troia *et al.*, "Identification of tidal-traffic patterns in metro-area mobile networks via matrix factorization based model," in *IEEE PerCom*, Mar 2017, pp. 297–301.

[32] TIM, "Big Data Challenge," 2014. [Online]. Available: https://dandelion.eu/datamine/open-big-data/

[33] Ericson, "Ericson Mobiliy Report," 2016. [Online]. Available: http://www.ericsson.com

[34] ENAiKOON, "Open Cell ID," 2016. [Online]. Available: http://opencellid.org/

[35] N. Cvijetic, "Optical network evolution for 5g mobile applications and sdn-based control," in *16th Int. Telecommun. Netw. Strategy Planning Symp. (Networks)*, Sept 2014, pp. 1–5.

[36] Huawei Technologies, "Technological Developments and Trends of Optical Networks," White Paper, 2016. [Online]. Available: http://www.huawei.com/

[37] B. Mukherjee, *Optical WDM Networks*. Springer-Verlag NY, Inc., 2006.

[38] M. Tornatore, G. Maier, and A. Pattavina, "Capacity versus availability trade-offs for availability-based routing," *J. Opt. Netw.*, vol. 5, no. 11, pp. 858–869, Nov 2006.

[39] C.-Y. Hong *et al.*, "Achieving high utilization with software-driven wan," in *ACM SIGCOMM Comp Commu Rev*, vol. 43, no. 4, 2013, pp. 15–26.

[40] X. Jin *et al.*, "Dynamic scheduling of network updates," in *ACM SIGCOMM Comp Commun Rev*, vol. 44, no. 4, 2014, pp. 539–550.

[41] H. H. Liu *et al.*, "zupdate: Updating data center networks with zero loss," in *ACM SIGCOMM Comp Commun Rev*, vol. 43, no. 4, 2013, pp. 411–422.