

Propagation of information on undirected dependency graphs for road traffic inference

Cyril Furtlehner* Yufei Han[†] Jean-Marc Lasgouttes[‡]

Victorin Martin[‡] Fabien Moutarde[†]

* *INRIA Saclay LRI, France*

[†] *ROBOTIC Lab, Mines Paris-Tech France*

[‡] *INRIA Paris-Rocquencourt, France*

In this paper we will review some properties of the “belief propagation” iterative map used to perform Bayesian inference in a distributed way. We use this algorithm as a starting point to address the inverse problem of encoding observation data into a probabilistic model, and focus on the situation when the data have many different statistical components, representing a variety of independent patterns. Asymptotic analysis reveals a connection with some Hopfield model. We then discuss the relevance of these results to the problem of reconstructing and predicting traffic states based on floating car data and show some experiments based on artificial and real data.

1. Introduction

The “belief propagation algorithm”, originated in the artificial intelligence community for inference problems on Bayesian networks.¹⁶ It is a non-linear iterative map which propagates information on a dependency graph of variables in the form of messages between variables. It has been recognised to be a generic procedure, instantiated in various domains like error correcting codes, signal processing or constraints satisfaction problems with various names depending on the context:¹¹ the forward-backward algorithm for Hidden Markov Model selection; the Viterbi algorithm; Gallager’s sum-product algorithm in Information theory. It has also a nice statistical physics interpretation, as a minimiser of a Bethe free energy²² or as a solver of TAP equations in the spin-glass context.¹⁰ As a noticeable development in the recent years, related to the connection with statistical physics is the emergence of a new generation of algorithms for solving difficult combinatorial problems, like the survey propagation algorithm¹³ for constraint satisfaction problems or the affinity propagation for clustering.³

The use we make of this algorithm in this work is twofold. Assuming a set of high dimensional data, in the form of sparse observations covering a finite fraction of segments in a traffic network, we wish to encode the dependencies

between the variables in a MRF in such a way that inference on this MRF with BP is optimal in some way. The paper is organised as follows: in Section 2 we introduce the BP algorithm and review some of its properties. In Section 3 we describe the traffic application and the inference model. In Section 4 we give a statistical physics analysis of this model. Finally in Section 5 we present some preliminary tests of the method.

2. The Belief Propagation Algorithm

We consider a set of discrete random variables $\mathbf{x} = \{\mathbf{x}_i, i \in \mathcal{V}\} \in \{1, \dots, q\}^{|\mathcal{V}|}$ obeying a joint probability distribution of the form

$$\mathcal{P}(\mathbf{x}) = \prod_{\mathbf{a} \in \mathcal{F}} \psi_{\mathbf{a}}(\mathbf{x}_{\mathbf{a}}) \prod_{i \in \mathcal{V}} \phi_i(\mathbf{x}_i), \quad (1)$$

where ϕ_i and $\psi_{\mathbf{a}}$ are factors associated respectively to a single variable \mathbf{x}_i and to a subset $\mathbf{a} \in \mathcal{F}$ of variables, \mathcal{F} representing a set of cliques. The $\psi_{\mathbf{a}}$ are called the “factors” while the ϕ_i are there by convenience and could be reabsorbed in the definition of the factors. This distribution can be conveniently graphically represented with a bi-bipartite graph, called the factor graph;¹¹ \mathcal{F} together with \mathcal{V} define the factor graph \mathcal{G} , which will be assumed to be connected. The set \mathcal{E} of edges contains all the couples $(\mathbf{a}, i) \in \mathcal{F} \times \mathcal{V}$ such that $i \in \mathbf{a}$. We denote $d_{\mathbf{a}}$ (resp. d_i) the degree of the factor node \mathbf{a} (resp. to the variable node i). The factor graph on the Figure 2.1.a corresponds for example to the following measure

$$p(\mathbf{x}_1, \dots, \mathbf{x}_6) = \frac{1}{Z} \psi_a(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \psi_b(\mathbf{x}_4) \psi_c(\mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6)$$

with the following factor nodes $\mathbf{a} = \{1, 2, 3\}$, $\mathbf{b} = \{4\}$ and $\mathbf{c} = \{3, 5, 6\}$.

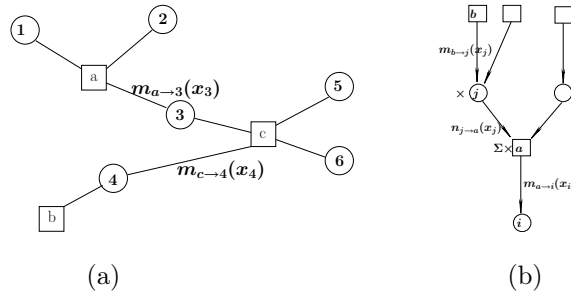


Fig. 2.1. Example of factor graph (a) and message propagation rules (b).

Assuming that the factor graph is a tree, computing the set of marginal distributions, called the belief $\mathbf{b}(\mathbf{x}_i = \mathbf{x})$ associated to each variable i can be done efficiently. The BP algorithm does this effectively for all variables in one single procedure, by remarking that the computation of each of these marginals involves intermediates quantities called the messages $\mathbf{m}_{\mathbf{a} \rightarrow i}(\mathbf{x}_i)$ [resp. $\mathbf{n}_{i \rightarrow \mathbf{a}}$] “sent” by factor node \mathbf{a} to variable node i [resp. variable node i to factor node \mathbf{a}], and which are necessary to compute other marginals. The idea of BP is

to compute at once all these messages using the relation among them as a fixed point equation. Iterating the following message update rules sketched on Figure 2.1.b:

$$\begin{cases} m_{a \rightarrow i}(x_i) \leftarrow \sum_{x_a \setminus x_i} \prod_{j \in a/i} n_{j \rightarrow a}(x_j) \psi_a(x_a) \\ n_{i \rightarrow a}(x_i) \leftarrow \phi_i(x_i) \prod_{b \ni i} m_{b \rightarrow i}(x_i) \end{cases}$$

yields, when a fixed point is reached, the following result for the beliefs,

$$b(x_i) = \frac{1}{Z_i} \phi_i(x_i) \prod_{a \ni i} m_{a \rightarrow i}(x_i)$$

$$b(x_a) = \frac{1}{Z_a} \psi_a(x_a) \prod_{i \in a} n_{i \rightarrow a}(x_i).$$

This is exact if the factor graph is a tree and only approximate on multiply connected factor graphs. As mentioned before, the set of beliefs which is obtained corresponds to a stationary point of a variational problem.²² Indeed, consider the Kullback-Leibler divergence between a test joint distribution $\mathbf{b}(\mathbf{x})$ with the reference one $\mathbf{p}(\mathbf{x})$. The Bethe approximation yield the following functional of the beliefs, including the joint beliefs $b_a(x_a)$ corresponding to each factor:

$$D_{KL}(\mathbf{b}, \mathbf{p}) = \sum_{\{x\}} b(\{x\}) \log \frac{b(\{x\})}{p(\{x\})}$$

$$\simeq \sum_{a, x_a} b_a(x_a) \log \frac{b_a(x_a)}{\prod_{i \in a} b_i(x_i) \psi(x_a)} + \sum_{i, x_i} \log \frac{b_i(x_i)}{\phi_i(x_i)}$$

$$\stackrel{\text{def}}{=} F_{\text{Bethe}} = E - S_{\text{Bethe}}$$

This is equivalent to say that we look for a minimiser of $D_{KL}(\mathbf{b}, \mathbf{p})$ in the following class of joint probabilities:

$$\mathbf{b}(\mathbf{x}) = \frac{1}{Z} \prod_a \frac{b_a(x_a)}{\prod_{i \in a} b_i(x_i)} \prod_i b_i(x_i)$$

under the constraint that

$$\sum_{x_a \setminus x_i} b_a(x_a) = b_i(x_i) \quad \forall a \in \mathcal{F}, \forall i \in a.$$

For a multi-connected factor graph, the beliefs are then interpreted as pseudo-marginal distribution, it is only when \mathcal{G} is simply connected that these are genuine marginal probabilities of the reference distribution \mathbf{p} .

There are a few properties of BP that are worth mentioning at this point. At first BP is a fast converging algorithm:

- Two sweeps over all edges are needed if the factor-graph is a tree.
- The complexity scales heuristically like $\mathbf{KN} \log(N)$ on a sparse factor-graph with connectivity \mathbf{K} .
- It is \mathbf{N}^2 for a complete graph.

However, when the graph is multiply connected, there is little guarantee on the convergence¹⁵ even so in practice it works well for sufficiently sparse graphs. Another limit in this case, is that the fixed point may not correspond to a true measure, so in this sense the obtained beliefs albeit compatible among each other are considered only as pseudo-marginals. Finally for such graphs the uniqueness of fixed points is not guaranteed, but what has been shown is that:

- stable BP fixed points are local minima of the Bethe free energy;⁸
- the converse is not necessarily true.¹⁸

There are two important special cases, where the BP equations simplify:

(i) For binary variables: $\mathbf{x}_i \in \{0, 1\}$. Upon normalisation the messages are parametrised as:

$$\mathbf{m}_{a \rightarrow i}(\mathbf{x}_i) = \mathbf{m}_{a \rightarrow i} \mathbf{x}_i + (1 - \mathbf{m}_{a \rightarrow i})(1 - \mathbf{x}_i),$$

which is stable w.r.t. the message update rule. Then the propagation of information reduces to the scalar quantity $\mathbf{m}_{a \rightarrow i}$.

(ii) For Gaussian variables, the factors are necessarily pairwise, of the form

$$\psi_{ij}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left\{-\frac{1}{2}[\mathbf{x}_i \mathbf{x}_j] \mathbf{A}_{ij} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{x}_j \end{bmatrix}\right\}$$

$$\phi_i(\mathbf{x}_i) = \exp\{h_i \mathbf{x}_i\}.$$

Since factors are pairwise, messages can then be considered to be sent directly from one variable node i to another j with a Gaussian form:

$$\mathbf{m}_{i \rightarrow j}(\mathbf{x}_j) = \exp\left(-\frac{(\mathbf{x}_j - \mathbf{x}_{ij})^2}{2\sigma_{ij}}\right).$$

This expression is also stable w.r.t. the message update rules. Information is then propagated via the 2-component real vector $(\mathbf{x}_{ij}, \sigma_{ij})$ with the following update rules:

$$\mathbf{x}_{ij} \leftarrow \frac{1}{A_{ij}} \sum_{k \in i \setminus j} \frac{\mathbf{x}_{ik}}{\sigma_{ik}} + h_i,$$

$$\sigma_{ij} \leftarrow \frac{1}{A_{ij}^2} \left[A_{ii} + \sum_{k \in i \setminus j} \frac{1}{\sigma_{ik}} \right].$$

In this case there is only one fixed point even on a loopy graph, not necessarily stable, but if convergence occurs the single variables beliefs provide the exact marginals.¹⁹ In fact for continuous variables, the Gaussian distribution is the only one compatible with the BP rules. Expectation propagation¹⁴ is a way to address more general distributions in an approximate manner.

3. Application context

3.1. Problem at hand

Once the underlying joint probability measure is given this algorithm can be very efficient for inferring hidden variables, but in real application it is often the case that we have first to build the model. This is precisely the case for the

application that we are considering concerning the reconstruction and prediction of road traffic conditions, typically on the secondary network from sparse observations. Existing solutions for traffic information are classically based on data coming from static sensors (magnetic loops) on main road axis. These devices are far too expensive to be installed everywhere in all streets on the traffic network and other sources of data have to be found. One recent solution comes from an increasing number of vehicles equipped with GPS and able to exchange data through cellular phone connections for example, able therefore to produce so-called Floating Car Data (FCD). Our objective in this context is to build an inference schema adapted to these FCD, able to run in real time and adapted to large scales road network, of size ranging from 10^3 to 10^5 in the number of segments. In this respect, the BP algorithm seems well suited, but the difficulty is to construct a model based on these FCD.

3.2. Statistical modelling and Inference schema

To set the inference schema, we assume that a large amount of FCD sent by probe vehicles concerning some area of interest are continuously collected over a reasonable period of time (one year or more) such as to allow a finite fraction (a few percents) of road segments to be covered in real time. Our approach is defined as follows:

- Historical FCD are used to compute empirical dependencies between contiguous segments of the road network.
- These dependencies are encoded into a graphical model which vertices are (segment, timestamps) pairs attached with a congestion state, i.e. typically CONGESTED/NOT-CONGESTED.
- Congestion probabilities of the various unvisited segments or in the short-term future are computed with BP, conditionally to real-time data.

On the factor-graph the information is propagated both temporally and spatially. In this perspective, reconstruction and prediction are taken on the same footing, even so we expect of course future prediction to be less precise than reconstruction.

This schema is based on a statistical description of traffic data which is obtained by spatial and temporal discretization, in terms of road segments i and discrete time slots t corresponding to time windows of typically a few minutes, leading to consider a set of vertices $\mathcal{V} = \{\alpha = (i, t)\}$. At each vertex is attached a microscopic degree of freedom $\mathbf{x}_\alpha \in \mathbf{E}$ being a descriptor of the corresponding segment state like for example, $\mathbf{E} = \{0, 1\}$, 0 for congested and 1 for fluid. The model itself is based on historical data in form of empirical marginal distributions $\hat{p}(\mathbf{x}_\alpha)$, $\hat{p}(\mathbf{x}_\alpha, \mathbf{x}_\beta)$ giving reference states and statistical interactions between degrees of freedom. Finally reconstruction and prediction is produced in the form of conditional marginal probability distribution $p(\mathbf{x}_\alpha | \mathcal{V}^*)$ of hidden variables $\mathcal{V} \setminus \mathcal{V}^*$ conditionally to the actual state of variables in the set $\mathcal{V}^* =$ of observed variables.

In addition to this microscopic view, it is highly desirable to enrich the description with macroscopic variables able in particular to capture and encode the temporal dynamics of the global system. These can be obtained by some linear analysis, e.g. PCA or with non-linear methods of clustering providing possibly hierarchical structures. Once some relevant variables are identified, we

can expect to have both a macroscopic description of the systems which can potentially be easily coupled to the microscopic one, by adding some nodes into the factor-graph. These additional degrees of freedom would be possibly interpreted in terms of global traffic indexes, associated to regions or components.

3.3. MRF model and pseudo moment matching calibration

At the microscopic level the next step is to define a MRF, on which to run BP with good inference properties. For the rest of the paper, we consider a binary encoding $\mathbf{x}_{i,t} \in \{0, 1\}$, which we interpret as congested/not-congested latent state, such that the MRF will actually be an Ising model as illustrated on Figure 3.1. We have to answer the two related questions:

- Given the historical data $\hat{p}(\mathbf{x}_{i,t}), \hat{p}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t'})$ which joint law $P(\{\mathbf{x}_{i,t}, (i, t) \in \mathcal{V}\})$?
- Given new observation $\{\mathbf{x}_{i,t}^*, (i, t) \in \mathcal{V}^*\}$ how to infer $\{\mathbf{x}_{i,t}, (i, t) \in \mathcal{V} \setminus \mathcal{V}^*\}$?

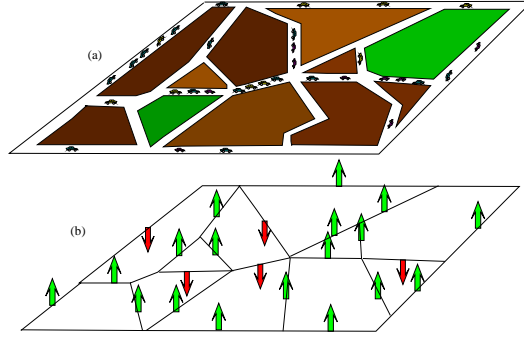


Fig. 3.1. Underlying Ising modelling of traffic configurations.

The solution that we have been exploring⁵ are based on mean-field techniques in statistical physics. It consists to use the Bethe approximation for the encoding and the belief-propagation for the decoding, such that the calibration of the model is coherent with the inference algorithm. In particular, when there is no real time observation, the reference point is given by the set of historical belief, so we want that running BP on our MRF delivers precisely these beliefs. Stated differently, we look for the ϕ and ψ defining the MRF in (1) such that the beliefs match the historical marginals:

$$\mathbf{b}_a(\mathbf{x}_a) = \hat{p}_a(\mathbf{x}_a)$$

There is an explicit solution to this problem, because BP is known to be coherent with the Bethe approximation,²² a consequence of what is that any BP fixed point \mathbf{b} has to verify

$$\mathcal{P}(\mathbf{x}) = \prod_{i \in \mathcal{V}} \phi_i(\mathbf{x}_i) \prod_{a \in \mathcal{F}} \psi_a(\mathbf{x}_a) = \prod_{i \in \mathcal{V}} \mathbf{b}_i(\mathbf{x}_i) \prod_{a \in \mathcal{F}} \frac{\mathbf{b}_a(\mathbf{x}_a)}{\prod_{i \in a} \mathbf{b}_i(\mathbf{x}_i)}. \quad (2)$$

As a result, a *canonical choice* for the functions ϕ and ψ is simply

$$\phi_i(\mathbf{x}_i) = \hat{p}_i(\mathbf{x}_i), \quad \psi_a(\mathbf{x}_a) = \frac{\hat{p}_a(\mathbf{x}_a)}{\prod_{i \in a} \hat{p}_i(\mathbf{x}_i)} \quad (3)$$

going along with $\mathbf{m}_{a \rightarrow i}(\mathbf{x}_i) \equiv \mathbf{1}$ as a particular BP fixed point. In addition, from the reparametrization property of BP,¹⁷ any other choices verifying (2) produces the same set of fixed points with the same convergence properties. Note that more advanced methods than the Bethe Approximation have been proposed, corresponding mainly to various levels of accuracy in the linear response theory, for matching pseudo-moments^{12,20,21} but at a higher computational price.

Next, for the decoding part, inserting information in real time in the model is done as follows. In practice observation are in the form of real numbers like speed or travel time. One possibility is to project such an observation onto the binary state $\mathbf{x}_i = \mathbf{0}$ or $\mathbf{x}_i = \mathbf{1}$ but in practice this proves to be too crude. Since the output of BP is anyway in the form of beliefs, i.e. real numbers in $[0, 1]$, the idea is to exploit the full information by defining a correspondence between observations \mathbf{Y}_i and probabilities $\mathbf{p}^*(\mathbf{x}_i = \mathbf{1})$. The optimal way of inserting this quantity into the BP equations,⁴ is obtained variationally by imposing the additional constraint $\mathbf{b}_i = \mathbf{p}^*(\mathbf{x}_i)$ which results in modified messages sent from $i \in \mathcal{V}^*$, now reading

$$n_{i \rightarrow a}(\mathbf{x}_i) = \frac{p_i^*(\mathbf{x}_i)}{m_{a \rightarrow i}(\mathbf{x}_i)} \quad \left[= \frac{p_i^*(\mathbf{x}_i)}{b_i(\mathbf{x}_i)} \prod_{a' \ni i, a' \neq a} m_{a' \rightarrow i}(\mathbf{x}_i) \right]$$

4. Statistical Physics Analysis

Some preliminary experiments of this procedure performed in⁵ indicate that many BP fixed point can exist in absence of information, each one corresponding to some congestion pattern like e.g. congestion/free flow. In⁶ we have analysed the presence of multiple fixed points by looking at a study case and we outline some of the results in this section. In this study we considered a generative hidden model of traffic in the form of a probabilistic mixture, $C \ll N$ being the number of mixture components, with each component having a simple product form:

$$P_{\text{hidden}}(\mathbf{x}) \stackrel{\text{def}}{=} \frac{1}{C} \sum_{c=1}^C \prod_{i \in \mathcal{V}} p_i^c(\mathbf{x}_i). \quad (4)$$

The interpretation of this model is that traffic congestion is organised in various patterns which can show up at different moments. We then studied the behaviour of our inference model on the data generated by this hidden probability by adding a single parameter α into its definition (3):

$$\phi_i(\mathbf{x}_i) = \hat{p}_i(\mathbf{x}_i), \quad \psi_{ij}(\mathbf{x}_i, \mathbf{x}_j) = \left(\frac{\hat{p}_{ij}(\mathbf{x}_i, \mathbf{x}_j)}{\hat{p}_i(\mathbf{x}_i) \hat{p}_j(\mathbf{x}_j)} \right)^\alpha$$

where \hat{p}_i and \hat{p}_{ij} are again the **1**- and **2**- variables frequency statistics that constitute the input of the model while (4) is assumed to be unknown. In addition we impose some sparsity in the factor graph with help of some selection procedure of the links to maintained a reduced mean connectivity \mathbf{K} . The typical numerical experiment we perform, given a configuration randomly

sampled from (4), is to reveal gradually the variables $\mathbf{x}_{\mathcal{V}^*}$ in a random order and compute conditional predictions for the remaining unknown variables. We then compare the beliefs obtained with the true conditional marginal probabilities $P(\mathbf{x}_i = \mathbf{x} | \mathbf{x}_{\mathcal{V}^*})$ computed with (4), using an error measure based on the Kullback-Leibler distance:

$$D_{KL} \stackrel{\text{def}}{=} \left\langle \sum_{\mathbf{x} \in \{0,1\}} b_i(\mathbf{x}) \log \frac{b_i(\mathbf{x})}{P(\mathbf{x}_i = \mathbf{x} | \mathbf{x}_{\mathcal{V}^*})} \right\rangle_{\mathcal{V}^*}.$$

A sample test shown on Figure 4.1.a indicates for example that on a system

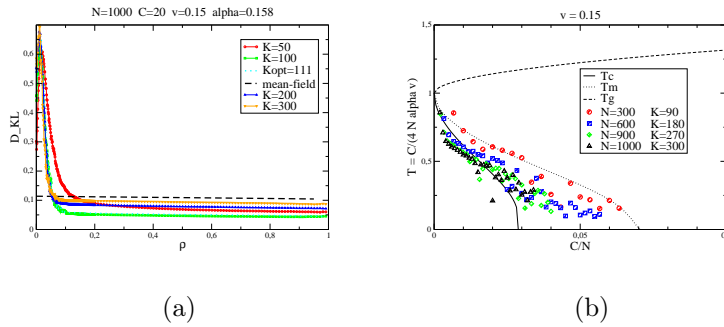


Fig. 4.1. D_{KL} error as a function of observed variables ρ (a). Phase diagram of the Hopfield model and optimal points found experimentally (b)

with 10^3 variables, it is possible with our model to infer with good precision a mixture of 20 components by observing 5% of the variables. To interpret these results letting $\mathbf{s}_i = 2\mathbf{x}_i - 1$, we first identify the Ising model corresponding to the MRF given by (1):

$$\mathcal{P}(\mathbf{s}) = \frac{1}{Z} e^{-\beta H[\mathbf{s}]},$$

with an inverse temperature β and the Hamiltonian

$$H[\mathbf{s}] \stackrel{\text{def}}{=} -\frac{1}{2} \sum_{i,j} J_{ij} s_i s_j - \sum_i h_i s_i.$$

The identification reads:

$$\beta J_{ij} = \frac{\alpha}{4} \log \frac{\hat{p}_{ij}(1,1)\hat{p}_{ij}(0,0)}{\hat{p}_{ij}(0,1)\hat{p}_{ij}(1,0)},$$

$$\beta h_i = \frac{1 - \alpha K_i}{2} \log \frac{\hat{p}_i(1)}{\hat{p}_i(0)} + \frac{\alpha}{4} \sum_{j \in \mathcal{I}} \log \frac{\hat{p}_{ij}(1,1)\hat{p}_{ij}(1,0)}{\hat{p}_{ij}(0,1)\hat{p}_{ij}(0,0)},$$

Then, in the limit $C \gg 1$, $N \gg C$ and fixed average connectivity K , we get asymptotically a mapping to the Hopfield model.⁹ The relevant parameters in this limit are $\eta = C/N$ and $v \in [0, 1/4]$ the variance of the variable bias

in the components. In this limit, the Hamiltonian is indeed similar to the one governing the dynamics of the Hopfield neural network model:

$$H[s] = -\frac{1}{2N} \sum_{i,j,c} \xi_i^c \xi_j^c s_i s_j - \sum_{i,c} h_i^c \xi_i^c s_i,$$

$$\text{with } \xi_i^c \stackrel{\text{def}}{=} \frac{p_i^c(1) - \frac{1}{2}}{\sqrt{v}} \quad \text{and} \quad h_i^c = \frac{C}{2\alpha K \sqrt{v}} - \frac{2C\sqrt{v}}{K} \sum_{j \in i} \text{Cov}(\xi_i^c, \xi_j^c),$$

with the inverse temperature given by

$$\beta = \frac{4\alpha v K}{C} \quad (\text{adapted to a non-complete graph})$$

Using mean-field methods, the phase diagram of this model has been established.¹ There are **3** phases separated by the transition lines T_g separating the paramagnetic phase from the spin glass phase and T_c separating the spin glass phase from the ferromagnetic phase (see Figure 4.1.b). The latter corresponding to the so-called *Mattis states*, i.e. to spin configurations correlated with one of the mixture component, of direct relevance w.r.t. inference. Locating the various models obtained in this diagram, depending on the parameter help to understand whether inference is possible or not with our MRF model.

5. Experiments with Synthetic and Real Data

Using both synthetic and real data we perform two kind of numerical tests:

- (i) reconstruction/prediction experiments
- (ii) automatic segmentation and BP fixed points identification

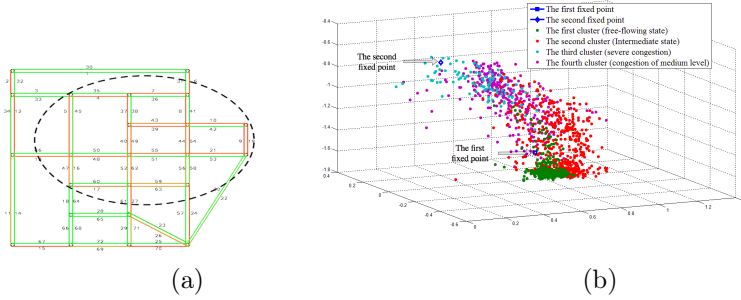


Fig. 5.1. Segmentation and BP fixed point identification (b) for Siouxfalls network (a).

In the first kind of experiments as in the preceding section, the fraction of observed variables ρ is varied and variables with time stamp in t_0 (present) or $t > t_0$ (future) are inferred, i.e. reconstructed or predicted respectively. In the type (ii) experiments, on one hand an automatic clustering of the data on reduced dimensional space is performed with machine learning techniques developed in.⁷ On the other hand the BP fixed points obtained at $\rho = 0$ are listed (see^{4,6} for details of the method) and compared to segmentation in the

reduced dimensional space. A first set of data has been generated with the traffic simulator “METROPOLIS”,² for a benchmark network called Siouxfalls shown on Figure 5.1.a. An example of the automatic clustering of spatial configurations with the corresponding BP fixed points associated to free flow and congestion is shown on Figure 5.1.b. To perform tests on real data we have also considered a dataset consisting of travel time measured every 3 minutes over 2 years of a Highway segmented into 62 segments. For this case, we refined the mapping between travel time and latent traffic variables by using for a given travel time tt the index being defined by:

$$u_i = f(tt) \stackrel{\text{def}}{=} P(tt_i < tt),$$

using for each segment $i = 1 \dots 62$ a weighted cumulative travel time distribution based on the automatic segmentation. On Figure 5.2.a is shown the result of this segmentation while on Figure 5.2.b is shown the result of a prediction with BP where error are computed on travel time.

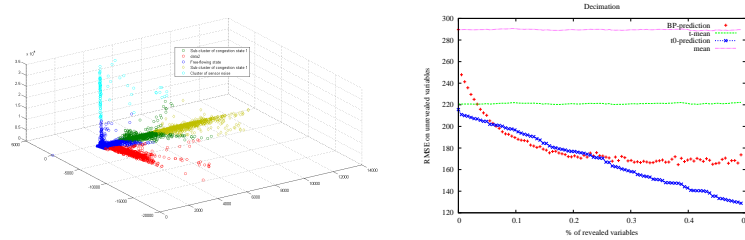


Fig. 5.2. (a) Automatic segmentation of Highway data. (b) BP prediction of $\mathbf{3}$ time layers in future as a function of the fraction of observed variables at t_0 . Comparison is made with a predictor combining recent available observations with historical time dependant mean.

6. Conclusion

In this paper we have reviewed some properties of the Belief-propagation algorithm and shown a promising approach for large scale applications in the context of traffic reconstruction and prediction. Concerning this application our method assume that traffic congestion is well represented by multiple distant pattern superposition which needs validation with real data on networks. Our reconstruction schema seems to work already with simple underlying binary indexes, but more work is needed for the dynamical part to be able to perform prediction.

References

1. AMIT, D. J., GUTFREUND, H., AND SOMPOLINSKY, H. Statistical mechanics of neural networks near saturation. *Annals of Physics* 173, 1 (1987), 30–67.
2. DE PALMA, A., AND MARCHAL, F. Real cases applications of the fully dynamic METROPOLIS tool-box: an advocacy for large-scale mesoscopic transportation systems. *Networks and Spatial Economics* 2, 4 (2002), 347–369.

3. FREY, B., AND DUECK, D. Clustering by passing messages between data points. *Science* 315 (2007), 972–976.
4. FURTLERHNER, C., HAN, Y., LASGOUTTES, J.-M., MARTIN, V., MARCHAL, F., AND MOUTARDE, F. Spatial and temporal analysis of traffic states on large scale networks. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on* (2010), pp. 1215–1220.
5. FURTLERHNER, C., LASGOUTTES, J., AND DE LA FORTELLE, A. A belief propagation approach to traffic prediction using probe vehicles. In *Proc. IEEE 10th Int. Conf. Intel. Trans. Sys.* (2007), pp. 1022–1027.
6. FURTLERHNER, C., LASGOUTTES, J.-M., AND AUGER, A. Learning multiple belief propagation fixed points for real time inference. *Physica A: Statistical Mechanics and its Applications* 389, 1 (2010), 149–163.
7. HAN, Y., AND MOUTARDE, F. Analysis of Network-level Traffic States using Locality Preservative Non-negative Matrix Factorization. In *Proc. of ITSC* (2011).
8. HESKES, T. On the uniqueness of loopy belief propagation fixed points. *Neural Computation* 16 (2004), 2379–2413.
9. HOPFIELD, J. J. Neural network and physical systems with emergent collective computational abilities. *Proc. of Natl. Acad. Sci. USA* 79 (1982), 2554–2558.
10. KABASHIMA, Y., AND SAAD, D. Belief propagation vs. tap for decoding corrupted messages. *Europhys. Lett.* 44 (1998), 668.
11. KSCHISCHANG, F. R., FREY, B. J., AND LOELIGER, H. A. Factor graphs and the sum-product algorithm. *IEEE Trans. on Inf. Th.* 47, 2 (2001), 498–519.
12. MEZARD, M., AND MORA, T. Constraint satisfaction problems and neural networks: A statistical physics perspective. *Journal of Physiology-Paris* 103, 1-2 (2009), 107 – 113.
13. MÉZARD, M., AND ZECCHINA, R. The random K-satisfiability problem: from an analytic solution to an efficient algorithm. *Phys.Rev.E* 66 (2002), 56126.
14. MINKA, T. Expectation propagation for approximate bayesian inference. In *Proceedings UAI* (2001), pp. 362–369.
15. MOOIJ, J. M., AND KAPPEN, H. J. On the properties of the Bethe approximation and loopy belief propagation on binary network. *J. Stat. Mech.* (2005), P11012.
16. PEARL, J. *Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference*. Morgan Kaufmann, 1988.
17. WAINWRIGHT, M. J. *Stochastic processes on graphs with cycles: geometric and variational approaches*. PhD thesis, MIT, Jan. 2002.
18. WATANABE, Y., AND FUKUMIZU, K. Graph zeta function in the bethe free energy and loopy belief propagation. In *Advances in Neural Information Processing Systems* (2009), vol. 22, pp. 2017–2025.
19. WEISS, Y., AND FREEMAN, W. T. Correctness of belief propagation in gaussian graphical models of arbitrary topology. *Neural Comput.* 13, 10 (2001), 2173–2200.
20. WELLING, M., AND TEH, Y. W. Approximate inference in boltzmann ma-

- chines. *Artif. Intell.* *143*, 1 (2003), 19–50.
21. YASUDA, M., AND TANAKA, K. Approximate learning algorithm in boltzmann machines. *Neural Comput.* *21* (2009), 3130–3178.
 22. YEDIDIA, J. S., FREEMAN, W. T., AND WEISS, Y. Generalized belief propagation. *Advances in Neural Information Processing Systems* (2001), 689–695.