PARALLEL DISTRIBUTED RELAXATION LABELING

S.Z. Li

School of EEE, Nanyang Technological University
Singapore 639798
szli@szli.eee.ntu.ac.sg

ABSTRACT

A parallel distributed relaxation labeling (RL) method, called the Lagrange-Hopfield (LH) method, is presented. RL is treated as a constrained optimization problem. The LH method solves the problem using the augmented Lagrangian multiplier technique and the graded Hopfield network. The LH method effectively overcomes instabilities that are inherent in the penalty method (*e.g.* Hopfield network) or the Lagrange multiplier method in constrained optimization. Due to the use of Lagrangian multipliers, the normalization operation in traditional RL methods is dispensed with. This makes the LH algorithm fully parallel and distributed and is suitable for analog implementation. Experiments also show that the method is able to produce good solutions in terms of the optimized objective values.

1. INTRODUCTION

Relaxation labeling (RL) [17] is a class of parallel iterative numerical procedures which use contextual constraints to reduce ambiguities in image analysis. It is widely used to solve image processing and vision problems where contextual constraints play an important role. The applications include image smoothing, enhancement, remote sensing, edge and boundary segmentation, motion analysis, perceptual organization and object matching and recognition.

Mathematically, RL can be posed as a problem of constrained optimization, as pointed out by Ullman [18]. Faugeras and Berthod [4] define a class of global criteria in terms of transition probabilities. Haralick [7] illustrates RL as minimizing the expected loss from a Bayes viewpoint. Hummel and Zucker [10] formulate the probabilistic relaxation approach as a global optimization problem in which a so called average local consistency measure is maximized.

There are various algorithms which can be used for computing RL solutions. Two basic types of RL algorithms are proposed in [17] for discrete and continuous updating of labeling assignment. A method known as gradient projection is used for the constrained optimization in [4, 10], which are shown in a comparison work [16] to yield good quality solutions. Techniques from mean field theory may be applied to obtain approximate solutions [13]. The above are deterministic RL algorithms which are local optimizers. A stochastic RL algorithm, simulated annealing (SA) [11, 5], is proven to converge a global optimum when the annealing schedule is slow enough. Disadvantages of SA are the high computational cost and difficulties in tuning the annealing schedule [1].

Although the continuous RL algorithms such as [17, 4, 10, 13] are parallelizable, they are not distributive. This is because of the requirement that the labeling assignment for each object to be labeled be normalized, *i.e.* summed into 1. The gradient projection in [4, 10] is actually a global operation.

This paper proposes a novel algorithm, called the Lagrange-Hopfield (LH) method, for continuous RL. It is based on the augmented Lagrange multiplier method for constrained optimization [15, 8] and the graded Hopfield method [9] for bridging the gap between discrete and continuous optimization. As an optimization network, the LH method effectively overcomes instabilities that are inherent in the penalty method (*e.g.* Hopfield network) or the Lagrange multiplier method. Owing to the use of Lagrangian multipliers, the normalization operation in traditional RL methods whereby the labeling assignment for each object is summed into one is dispensed with. The resulting algorithm solves a system of unconstrained differential equations. This makes the LH algorithm fully parallel and distributed and is suitable for analog implementation.

Experiments are done to compare the LH algorithm with the deterministic RL algorithms of [17, 10, 13] and the SA algorithm of [11]. The results show that the LH algorithm yields good quality solutions comparable to produced by simulated annealing.

The rest of the paper is organized as follows: Section 2 introduces the notations for RL. Section 3 presents the LH method. Section 4 describes the experiments. Section 5 gives conclusions.

2. CONTINUOUS RELAXATION LABELING

Let $\mathcal{S} = \{1, \ldots, m\}$ be a set of m sites (termed objects in the traditional RL literature) to be labeled and $\mathcal{L} = \{0, 1, \ldots, M\}$ be a set of $M+1$ labels to be assigned to the sites. In continuous RL, the labeling state for each site $i \in \mathcal{S}$ is represented by an $M+1$ position vector

$$p_i = [p_i(I) \mid I \in \mathcal{L}] \tag{1}$$

subject to the following *feasibility* constraints

$$\begin{cases} \sum_{I \in \mathcal{L}} p_i(I) = 1 & \forall i \\ p_i(I) \geq 0 & \forall i, I \end{cases} \tag{2}$$

The real value $p_i(I) \in [0,1]$ reflects the strength with which i is assigned label I. Each p_i lies in a hyperplane in the non-negative quadrant portion of the multi-dimensional real space $\mathbb{R}^{\mathcal{L}}$. The set $p = \{p_i \mid i \in \mathcal{S}\}$ is called a *labeling assignment*. The feasible space for labeling assignments is

$$\mathbb{P} = \{p \mid p_i(I) > 0, \sum_{I \in \mathcal{L}} p_i(I) = 1, i \in \mathcal{S}, I \in \mathcal{L}\} \tag{3}$$

However, the final solution p^* must be *unambiguous*, that is,

$$p_i^*(I) = \{0, 1\} \qquad \forall i, I \tag{4}$$

with $p_i^*(I) = 1$ meaning that i is unambiguously labeled I. Therefore, the space for the final labeling assignments is

$$\mathbb{P}^* = \{p^* \mid p_i(I) \in \{0,1\}, \sum_{I \in \mathcal{L}} p_i(I) = 1, i \in \mathcal{S}, I \in \mathcal{L}\} \tag{5}$$

$\mathbb{P}^*$ is the set of "corners" of $\mathbb{P}$. Non-corner points in the continuous space $\mathbb{P}$ provide routes to the corners, *i.e.* the points in $\mathbb{P}^*$.

The continuous RL can be regarded as an iterative process for minimizing the following energy

$$\begin{aligned} E(p) \quad = \quad & \sum_{i \in \mathcal{S}} \sum_{I \in \mathcal{L}} r_i(I)\ p_i(I) + \qquad (6) \\ & \sum_{i \in \mathcal{S}} \sum_{I \in \mathcal{L}} \sum_{i' \in \mathcal{S}, i' \neq i} \sum_{I' \in \mathcal{L}} r_{i,i'}(I, I')\ p_i(I)\ p_{i'}(I') \end{aligned}$$

where $r_i(I)$ and $r_{i,i'}(I, I')$ are unary and binary compatibility functions. The minimization is over $\mathbb{P}$ and finally over $\mathbb{P}^*$ and hence is constrained. The constrained minimization is done based on the gradient $q = \nabla E(p) = \{q_i(I) \mid \forall (i, I)\}$ where the gradient components are given by

$$q_i(I) = \frac{\partial E}{\partial p_i(I)} = r_i(I) + 2 \sum_{i' \in \mathcal{S}, i' \neq i} \sum_{I' \in \mathcal{L}} r_{i,i'}(I, I')\ p_{i'}(I') \tag{7}$$

for symmetric $r_{i,i'}(I, I')$. The initial assignment is set to $p^{(0)} \in \mathbb{P}$. The current $p^{(t)}$ is updated into a new state $p^{(t+1)}$ according to

$$p^{(t+1)} \leftarrow \Phi(p^{(t)}, q^{(t)}) \tag{8}$$

where Φ is an updating operator. RL algorithms differ from one another in the choice of Φ.

3. LAGRANGE-HOPFIELD METHOD

Let us state the RL problem more formally as the following constrained optimization:

$$\begin{aligned} \min_f \qquad & E(p) & (9) \\ \text{subject to} \qquad & C_i(p) = 0 \quad \forall i \in \mathcal{S} & (10) \\ & p_i(I) \geq 0 \quad \forall i \in \mathcal{S}, I \in \mathcal{L} & (11) \end{aligned}$$

where

$$C_i(p) = \sum_{I \in \mathcal{L}} p_i(I) - 1 \tag{12}$$

Equation (10) and (11) defines the feasible space $\mathbb{P}$. The final solution must be unambiguous:

$$p_i^*(I) \in \{0, 1\} \tag{13}$$

The LH method for solving the above problem is described in the following.

First, we introduce internal variables $u_i(I) \in (-\infty, +\infty)$ $(\forall i, I)$ and relate them to $p_i(I)$ via

$$p_i(I) = \psi_T(u_i(I)) \tag{14}$$

where $\psi_T(x)$ is a sigmoid function

$$\psi_T(x) = 1/[1 + e^{-x/T}] \tag{15}$$

controlled by a temperature parameter $T > 0$. With the introduction of the u variables, the energy function can be considered as $E(u) = E(p(u))$. This treatment confines $p_i(I)$ to the range $(0,1)$ and imposes the constraints of (11). In very high gain when $T \to 0^+$, $p_i(I)$ is forced to be 0 or 1 depending on whether $u_i(I)$ is positive or negative, thus imposing the constraints of (13).

Next, we use the Lagrange multiplier method to impose the constraints of (10). Define the following Lagrange function

$$L(p,\gamma) = E(p) + \sum_{i\in\mathcal{S}} \gamma_i C_i(p) \tag{16}$$

where γ_i are called Lagrange multipliers. It is a function of the $M \times m$ variables of p and m variables of γ. Penalty terms $[C_i(p)]^2$ can be added to give an augmented Lagrange function [15, 8]

$$L(p,\gamma) = E(p) + \sum_i \gamma_i C_i(p) + \beta \sum_i [C_i(p)]^2 \tag{17}$$

where $\beta > 0$ is a finite weight. Our experiments show that the penalty terms are generally necessary for damping oscillations and ensuring the convergence for the algorithm described below.

For p^* to be a local minimum subject to the constraints, it is necessary that (p^*, γ^*) be a stationary point of the Lagrange function:

$$\begin{array}{c} \nabla_p L(p^*,\gamma^*) = 0 \\ \nabla_\gamma L(p^*,\gamma^*) = 0 \end{array} \tag{18}$$

If (p^*, γ^*) is a saddle point for which

$$L(p^*,\gamma) \le L(p^*,\gamma^*) \le L(p,\gamma^*) \tag{19}$$

then p^* is a local minimum of $E(p)$ satisfying $C_k(p^*) = 0$ [6].

The following dynamics for the p variables can be used to find such a saddle point

$$\begin{aligned} \frac{\mathrm{d}p_i(I)}{\mathrm{d}t} &= -\frac{\partial L(p,\gamma)}{\partial p_i(I)} \qquad (20) \\ &= -\frac{\partial E(p)}{\partial p_i(I)} - \gamma_i \frac{\partial C_i(p)}{\partial p_i(I)} - \beta C_i(p) \frac{\partial C_i(p)}{\partial p_i(I)} \\ \frac{\mathrm{d}\gamma_i}{\mathrm{d}t} &= +\frac{\partial L(p,\gamma)}{\partial \gamma_i} = +C_i(p) \qquad (21) \end{aligned}$$

where $\frac{\partial E(p)}{\partial p_i(I)} = q_i(I)$ is defined in (7) and $\frac{\partial C_i(p)}{\partial p_i(I)} = 1$ for $C_i(p) = \sum_{I\in\mathcal{L}} p_i(I) - 1$. It performs energy descent on p but ascent on γ. The convergence of this system has been obtained by [2]. Its application in relaxation labeling is suggested by [18] without using the u variables. It has also been used for solving the traveling salesman problem [14, 19].

In the LH method, the updating of the labeling variables is performed on u rather than on p. Equ.(20) is replaced by

$$\begin{aligned} \frac{\mathrm{d}u_i(I)}{\mathrm{d}t} &= -\frac{\partial L(p(u),\gamma)}{\partial p_i(I)} \frac{\partial p_i(I)}{\partial u_i(I)} \qquad (22) \\ &= -\left[q_i(I)/m + \gamma_i + \beta C_i(p)\right] \frac{\partial p_i(I)}{\partial u_i(I)} \end{aligned}$$

where a $q_i(I)$ is weighted by a factor of $1/m$ to achieve some invariance to m. Note that $\frac{\partial p_i(I)}{\partial u_i(I)} = \frac{\mathrm{e}^{-u_i(I)/T}}{T(1+\mathrm{e}^{-u_i(I)/T})^2}$ is always positive.

A damping term $\zeta u_i(I)$, where $\zeta > 0$, may be added to (22) as in the graded Hopfield neural networks [9]. This is equivalent to adding an energy term

$$H(p) = \sum_i \sum_I \int_0^{p_i(I)} \psi_T^{-1}(p)\mathrm{d}p \tag{23}$$

to $L(p(u),\gamma)$, where the inverse of the sigmoid function ψ_T is

$$\psi_T^{-1}(p_i(I)) = u_i(I) = T \ln \frac{p_i(I)}{1-p_i(I)} \tag{24}$$

The term $H(p)$ reaches the minimum of zero only when all $p_i(I)$'s are either 0 or 1. This means minimizing this term with $T \to 0^+$ in effect leads to the satisfaction of (13). However, this term is not necessary in our model because the satisfaction of (10) and (11) under a small T for $p_i(I) = 1/[1+e^{-u_i(I)/T}]$ is enough for the convergence to (13). The competitive mechanism in (11) will make the winner take all.

In summary, the LH algorithm implements the three equations, (21), (14) and (22). The updating consists of the three steps:

$$\begin{aligned} &u_i(I) \leftarrow u_i(I) - \\ &\mu \tfrac{\partial p_i(I)}{\partial u_i(I)} \left\{q_i(I)/m + \gamma_i + \beta \left[\textstyle\sum_I p_i(I) - 1\right]\right\} \end{aligned} \tag{25}$$

where μ is a step size factor

$$p_i(I) \leftarrow \psi_T(u_i(I)) \tag{26}$$

and

$$\gamma_i \leftarrow \gamma_i + \lambda \left[\sum_I p_i(I) - 1\right] \tag{27}$$

where T is decreasing and λ increasing. The above is formulated for minimizing the energy function $E(p)$. To maximize a function $G(f)$ with the constraints, we can simply let $E(p) = -G(p)$ and then apply the same updating equations.

The above LH algorithm represents an unconstrained system and it is in contrast to the existing RL algorithms. RL needs a normalization process or alike to meet the feasibility condition. In [17], for example, Rosenfeld *et al.* use the following fixed point iteration

$$p_i^{(t+1)}(I) \leftarrow \frac{p_i^{(t)}(I)(1+q_i^{(t)}(I))}{\sum_I p_i^{(t)}(I)(1+q_i^{(t)}(I))} \tag{28}$$

for maximizing E(p). Other improved algorithms involve more sophisticated operations such as gradient projection [4, 10] and mean field annealing [13]. These are dispensed with in LH due to the use of the Lagrange multipliers. The way that LH works make it suitable for distributive implementation.

4. EXPERIMENTS

In the following experiments, we compare several RL algorithms in terms of the solution quality measured by the maximized gain $G(p^*)$, the computational cost measured by the number of required iterations, and the need for heuristics for tuning the algorithm. The compared algorithms are

1. ICM [3]
2. RL of [10]
3. RL of [17]
4. MFA of [13]
5. SA of [5]
6. The LH method

The test-bed for the comparison is relational matching. The MAP-MRF formulation described in [12] is used to define the compatibility matrix $[r_{i,i'}(I,I')]$. The parameters in Equations (25) – (27) are set as follows: $\beta = 100, T$ decreasing from 100000 to not smaller than 10000 according to $T \leftarrow 0.999T$, λ increasing from 1 to 100 according to $\lambda \leftarrow 1.01\lambda$, and $\mu \frac{\partial p_i(I)}{\partial u_i(I)} = 1000$ is constant for the sequential implementation. The variables are initialized as $u_i(I) = 0$ and $\gamma_i = 0$. Figures 1 and 2 demonstrate the solution quality in terms of the maximized gain and the cost in terms the number of iterations, respectively. Every result is due to the average of 200 runs.

In terms of the solution quality, the proposed RL algorithm performs almost as well as the benchmarking SA algorithm and the best of the existing RL algorithms. In terms of the iteration number, it is several order lower that for SA. Further, it is much simpler than SA in terms of the ease in the tuning of parameters and schedules.

5. CONCLUSION

A novel relaxation labeling (RL) method, called the Lagrange-Hopfield (LH) method, has been proposed. The stabilities of the LH are due to the use of the augmented Lagrange technique. Basically, one is converting a combinatorial optimization to a constrained real optimization and in turn to an unconstrained real optimization. In the last form, the energy for the unconstrained problem is a weighted sum of the original cost function and other terms for the constraints. To produce valid solutions, the relative weight for the original cost function must be kept small to verify the constraints; therefore the solutions are significantly biased from the true minimum of the original cost function, owing to large influence from the extra terms. With the use of Lagrange multipliers as in the augmented Lagrangian, the weighting values for the penalty terms can be much smaller than those required by the penalty method; thus the relative weight for the original objective is increased, which helps yielding a lower objective value. At the same time, the augmented Lagrange method also overcomes the zigzagging problem with the standard Lagrange method and improve the convergence. These advantages of the augmented Lagrangian plus the use of Hopfield method for imposing the inequality and unambiguity constraints have made the success of the LH method.

The LH is comparable to the best of the existing RL algorithms in terms of solution quality and convergence rate. Its advantage over the existing RL algorithms is that it avoids the needs for the normalization and/or gradient projection operations. The dynamic system for the computation of LH solutions is suitable for parallel distributed implementation using analog circuits. Such hardware is useful not only for problems which can be formulated as relaxation labeling such as edge labeling and object matching; but also for general combinatorial problems such as traveling sales-man problem, pattern clustering and perceptual grouping.

6. REFERENCES

[1] E. H. L. Aarts. *Simulated annealing and Boltzmann machines : a stochastic approach to combinatorial optimization and neural computing.* Wiley, 1989.

[2] K. J. Arrow, L. Hurwicz, and H. Uzawa. *Studies in Linear and Nonlinear Programming.* Stanford University Press, 1958.

[3] J. Besag. “On the statistical analysis of dirty pictures” (with discussions). *Journal of the Royal Statistical Society, Series B*, 48:259–302, 1986.

[4] O. D. Faugeras and M. Berthod. “Improving consistency and reducing ambiguity in stochastic labeling: An optimization approach”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:412–423, April 1981.

[5] S. Geman and D. Geman. “Stochastic relaxation, Gibbs distribution and the Bayesian restoration of

images". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, November 1984.

[6] B. S. Gottfried. *Introduction to Optimization Theory*. Prentice-Hall, 1973.

[7] R. M. Haralick. "Decision making in context". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(4):417–428, July 1983.

[8] M. R. Hestenes. "Multipler and gradient methods". *Journal of Optimization Theory and Applications*, 4:303–320, 1969.

[9] J. J. Hopfield. "Neurons with graded response have collective computational properties like those of two state neurons". *Proceedings of National Academic Science, USA*, 81:3088–3092, 1984.

[10] R. A. Hummel and S. W. Zucker. "On the foundations of relaxation labeling process". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(3):267–286, May 1983.

[11] S. Kirkpatrick, C. D. Gellatt, and M. P. Vecchi. "Optimization by simulated annealing". *Science*, 220:671–680, 1983.

[12] S. Z. Li. "A Markov random field model for object matching under contextual constraints". In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 866–869, Seattle, Washington, June 1994.

[13] C. Peterson and B. Soderberg. "A new method for mapping optimization problems onto neural networks". *International Journal of Neural Systems*, 1(1):3–22, 1989.

[14] J. C. Platt and A. H. Barr. "Constrained differential optimization". In *Proceedings of the IEEE 1987 NIPS conference*, 1988.

[15] M. J. D. Powell. "A method of nonlinear constraints in minimization problems". In R. Fletcher, editor, *Optimization*, London, 1969. Academic Press.

[16] K. E. Price. "Relaxation matching techniques – A comparison". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(5):617–623, September 1985.

[17] A. Rosenfeld, R. Hummel, and S. Zucker. "Scene labeling by relaxation operations". *IEEE Transactions on Systems, Man and Cybernetics*, 6:420–433, June 1976.

[18] S. Ullman. "Relaxation and constraint optimization by local process". *Computer Graphics and Image Processing*, 10:115–195, 1979.

[19] E. Wacholder, J. Han, and R. C. Mann. "A neural network algorithm for the multiple traveling salesman problem". *Biological Cybernetics*, 61:11–19, 1989.

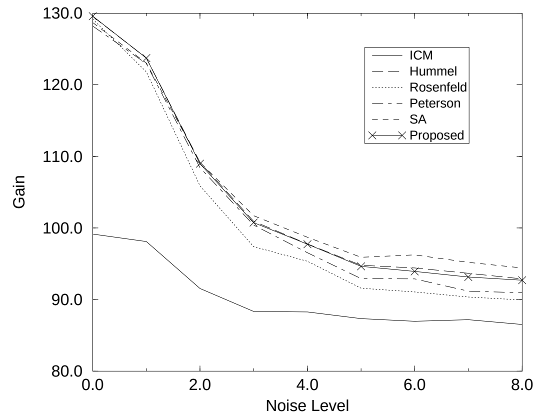

Figure 1: The maximized gain $G(p^*)$ (after divided by a factor of m^2). The higher, the better.

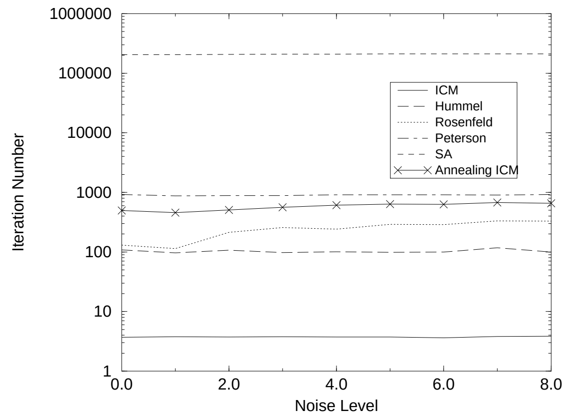

Figure 2: The number of iterations. The lower, the better.