Available online at www.sciencedirect.com

ScienceDirect

Computers & Operations Research 34 (2007) 2553–2561

computers & operations research

www.elsevier.com/locate/cor

Warm start by Hopfield neural networks for interior point methods

Marta I. Velazco Fontova[a,*], Aurelio R.L. Oliveira[b], Christiano Lyra[a]

[a]Faculty of Electrical Engineering and Computer Science (FEEC), Department of Systems Engineering (DENSIS), State University of Campinas (UNICAMP), Av. Albert Einstein 400, CP 6101, 13083-970, Campinas, SP, Brazil

[b]Institute of Mathematics, Statistics and Scientific Computing (IMECC), State University of Campinas (UNICAMP), Praça Sérgio Buarque de Holanda 651, CP 6065, 13083-859, Campinas, SP, Brazil

Available online 16 November 2005

Abstract

Hopfield neural networks and interior point methods are used in an integrated way to solve linear optimization problems. The Hopfield network gives warm start for the primal–dual interior point methods, which can be way ahead in the path to optimality. The approaches were applied to a set of real world linear programming problems. The integrated approaches provide promising results, indicating that there may be a place for neural networks in the “real game” of optimization.

Keywords: Hopfield networks; Primal–dual interior point methods; Neural networks; Linear programming

1. Introduction

This paper explores possibilities of cooperation between Hopfield neural networks and the “primal–dual family” of interior point methods [1–3] to solve linear optimization problems (i.e., linear programming problems).

The Hopfield network gives advanced starting points for the primal–dual interior point methods. The main motivation for this investigation is the observation that Hopfield neural networks can make significant advances in the earlier stages of optimization procedures but slow down significantly as they get close to the optimal solution [4]; interior point methods have the complementary characteristic of benefiting from warm starts and unveiling good directions towards optimality. To make things even more attractive the algebraic systems that appear in Hopfield neural networks and interior point methods are similar, allowing easy exchange of information and solution techniques.

Computing an initial advanced point or warm start for interior point methods is a research topic for many authors. Some approach it computing a solution of a perturbed problem with small modifications on b and/or c (sometimes also in A) [5]. The difficulty of this approach is that the optimal solution of the preceding problem is at or near the boundary of the feasible set, making it bad starting points for interior point methods—the methods take a long sequence of iterations because the points cannot get away from the boundary [6]. Gondzio [7,8] presents an approach with cutting plane methods; in this case the problem size increases and that can be a difficulty for large-scale problems.

Two approaches for cooperation between Hopfield networks and primal–dual interior point methods are investigated in the paper: with classical primal–dual and with predictor–corrector methods. In both approaches, Hopfield networks

* Corresponding author.

E-mail addresses: velazco@densis.fee.unicamp.br (M.I.V. Fontova), aurelio@ime.unicamp.br (A.R.L. Oliveira), chrlyra@densis.fee.unicamp.br (C. Lyra).

0305-0548/$ - see front matter
doi:10.1016/j.cor.2005.09.019

strive to boost starting points, passing the baton to the interior point methods way ahead in the paths to optimality. Applications to a set of real-life linear programming problems give guidelines on the possibilities of the approaches.

This paper is organized as follows. Section 2 introduces the primal–dual interior point methods for linear optimization. Section 3 reviews the Hopfield approach to solve optimization problems. Improvements in the Hopfield approach are presented in Section 4. The combined approaches are described in Section 5 and case studies are presented in Section 6. Conclusions follow.

2. The primal–dual interior point methods

This paper focuses on the infeasible "primal–dual" family of interior point methods: classical primal–dual method [1] and Predictor–Corrector [2,3]. Those methods resolve the primal and dual problem simultaneously using an initial primal–dual point not necessarily feasible. They are obtained by the application of Newton method to the optimality conditions.

Following [9] the primal linear optimization problem in the "standard" form can be stated as,

$$\begin{array}{ll} \min & c^T \cdot x \\ \text{s.t.} & A \cdot x = b, \quad x \geqslant 0, \end{array} \tag{1}$$

where $A \in \Re^{m\times n}$.

Associated to problem (1), there is the dual problem:

$$\begin{array}{ll} \max & b^T \cdot y \\ \text{s.t.} & A^T \cdot y + z = c, \quad z \geqslant 0. \end{array} \tag{2}$$

The following outlines a framework for such methods.

Given y^0 and $(x^0, z^0) > 0$ (an interior point)

For $k = 0, 1, 2, \ldots,$ Do

1. Choose $\sigma^k \in [0, 1)$ and set $\mu^k = \sigma^k \cdot (\gamma^k/n)$ where, $\gamma^k = (x^k)' \cdot z^k$ and n is the size of x.
2. Compute the Newton search directions, $(\Delta x^k, \Delta y^k, \Delta z^k)$.
3. Choose an appropriate step length to remain interior, $\alpha^k = \min(1, \tau^k \rho_p^k, \tau^k \rho_d^k)$, where $\tau^k \in (0, 1)$,

$$\rho_p^k = \min_{\{j:\Delta x_j^k<0\}} \left(\frac{-x_j^k}{\Delta x_j^k}\right) \quad \text{and} \quad \rho_d^k = \min_{\{j:\Delta z_j^k<0\}} \left(\frac{-z_j^k}{\Delta z_j^k}\right).$$

4. Compute the new interior point,

$$(x^{k+1}, y^{k+1}, z^{k+1}) = (x^k, y^k, z^k) + \alpha^k \cdot (\Delta x^k, \Delta y^k, \Delta z^k).$$

3. Modified Hopfield approach to linear optimization

The modified Hopfield network (MHN) proposed by da Silva [4] is a cycle of two phases: a feasibility stage and an updating stage—Fig. 1 illustrates the whole process. The feasibility stage (upper box in Fig. 1) screens the vector x and, if necessary, turns it into a feasible point. The updating stage (lower box in Fig. 1) computes a new vector x along a search direction. The process stops when it reaches a minimum.

3.1. Feasibility process

The feasibility process performs, iteratively, a cycle of two steps:

1. *Projection into the valid subspace* [10]: Represented by the box I.1 in Fig. 1, the projection computes a new feasible point x^p,

$$x^p = T \cdot x + s, \tag{4}$$

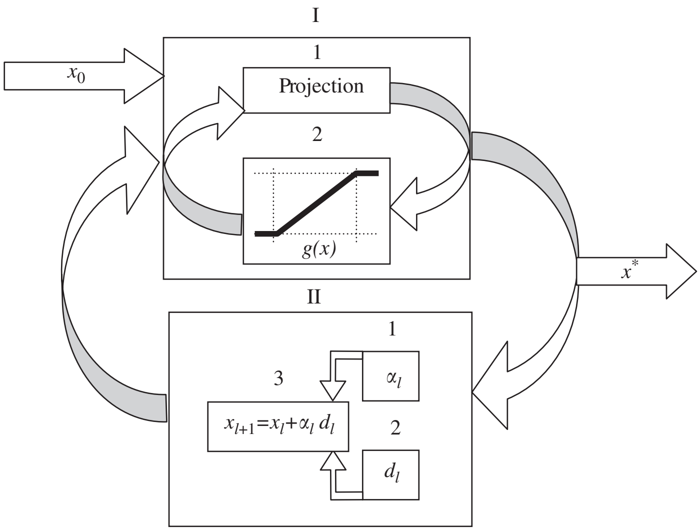

Fig. 1. Modified Hopfield network.

where T is a projection matrix that represents connections of neurons and s is orthogonal to T. Both T and s are computed from the constraints of the original optimization problem [10].

2. *Application of the activation function*: The activation function $(g(x))$, represented by the box I.2 in Fig. 1, pushes the point x into a feasible hypercube. Mathematically,

$$g(x) = \begin{cases} l, & x \leqslant l, \\ x, & l < x < u, \\ u, & x > u, \end{cases} \tag{5}$$

where l is the lower bound and u is the upper bound for variable x. Following the standard form of the primal linear optimization problem (Eq. (1)); $l = 0$.

3.2. Updating process

The feasible variable x is updated with the following sequence of steps [4], represented by box II in Fig. 1.

1. Compute the search direction, d^l.
2. Obtain the step size α_l with a fuzzy controller.
3. Compute the new vector (x^{l+1}),

$$x^{l+1} = x^l + \alpha^l \cdot d^l. \tag{6}$$

The search direction is computed from an energy function associated to the original problem. For linear problems in the standard form (1), the search direction is equal to the cost vector c during the whole iterative processes ($d^l = c$).

3.3. Discussion

It has been observed in case studies [4] that the convergence of the MHN is slow, even for very small instances. We consider that the main reasons for this behavior are associated to the following points:

1. The floating-point operations to carry on the projection process get too expensive as the size of the problem grows—Cholesky factorization [11] and know-how on the solution of sparse linear systems can provide paths for improvements;

2. The search direction, $d^l = c$, is fixed—recent developments in interior point methods can provide better directions;
3. The fuzzy controller obtains the step size with (basically) a cut-and-try approach—this is a costly procedure that grasps no significant information to speed up the iterative processes.

Next section discusses improvements on the MHN (including the points already outlined here).

4. Improvements on the modified Hopfield approach

This paper proposes new approaches to give a warm start for interior point method based on the cooperation between MHN and interior point methods (specifically, classical primal–dual and predictor–corrector methods). In order to implement those integrated approaches, modifications and enhancements are introduced in the MHN, with ideas borrowed from the interior point optimization area;

1. The projection process is enhanced;
2. The feasibility process is modified to produce a feasible interior point;
3. Interior point concepts provide better search directions [1–3];
4. A fixed step size is adopted (the fuzzy controller is eliminated);
5. In order to enable cooperation with the primal–dual methods, the MHN is tailored to provide a feasible dual interior point (the MHN provided only a primal feasible interior point).

All these points will be further discussed.

4.1. Enhancing efficiency of the projection process

The matrix T and vector s for primal linear optimization problems are computed by the following equations [12].

$$T = I - A^T \cdot (A \cdot A^T)^{-1} \cdot A, \tag{7}$$

$$s = A^T.(A \cdot A^T)^{-1} \cdot b, \tag{8}$$

where I is the identity matrix and A and b are given by (1).

Matrix T does not need to be explicitly computed—since A has few nonzero entries and a large number of floating-point operations are required when T is computed. The equations to obtain matrix T and the vector s have some algebraic operations with the matrix A that can be performed before the projection process. The projected vector x^p can be computed using (4), (7) and (8), as follows:

$$x^p = (I - A^T \cdot (A \cdot A^T)^{-1} \cdot A) \cdot x + A^T.(A \cdot A^T)^{-1} \cdot b, \tag{9}$$

Eq. (9) can be rewritten as,

$$x^p = x + (A^T \cdot (A \cdot A^T)^{-1}) \cdot (b - A \cdot x). \tag{10}$$

The Cholesky factorization of $A \cdot A^T$ [11], L_R, can be used—it needs to be computed only once, before starting the iterating process. Therefore, the new projected vector is computed as follows:

$$x^p = x + A^T \cdot (L_R^{-T} \cdot (L_R^{-1} \cdot (b - A \cdot x))), \tag{11}$$

$$L_R \cdot L_R^T = A \cdot A^T. \tag{12}$$

4.2. The feasibility process

To simplify the presentation, this paper does not consider upper bounds for variables in the optimization problems (in other words, the upper bound u is assumed to be infinite)—generalization to problems with upper bounds is trivial. Therefore, the feasible hypercube ($0 \leqslant x \leqslant u$) is simplified into $0 \leqslant x$. Also, a tolerance (*It*) is introduced to guarantee that x is an interior point (in other words, that $x > 0$).

The tolerance (It) is used in the definitions of the activation function ($g_{It}(x)$) of box I (Fig. 1), as follows:

$$g_{It}(x_i) = \begin{cases} It, & x_i < It, \\ x_i, & x_i \geqslant It. \end{cases} \tag{13}$$

Note that this new function ($g_{It}(x)$) modifies Step 2 of the feasibility process in MHN, giving points that are interior.

4.3. Search direction

The original search direction is related to the steepest descent method, which gives linear convergence. The proposed interior point search direction comes from the Newton's method, which achieves quadratic convergence [9].

4.4. Step size

A fuzzy controller was adopted by da Silva [4] to compute the step used in the MHN. For the Hopfield-interior point approach the fuzzy controller is of no use. The fuzzy controller is eliminated and a fixed step size is used.

A good guess for the step size is to use 100% [1–3] of the search direction. Following this path, the step size used in the Hopfield-interior point approaches is $\alpha = 1$. Moreover, the feasibility process keeps the point interior to the feasible region.

4.5. Computing a feasible dual interior point

The dual problem (2) can be written as

$$\begin{aligned} \max \quad & b^T \cdot y \\ \text{s.t.} \quad & [A^T \ \ I_n] \cdot \begin{bmatrix} y \\ z \end{bmatrix} = c, \quad z \geqslant 0, \end{aligned} \tag{14}$$

where I_n is the $n \times n$ identity matrix. Using this result in (7) and (8), matrix T and vector s can be computed by the following equations:

$$T = I_{m+n} - [A^T \ \ I_n]^T \cdot ([A^T \ \ I_n] \cdot [A^T \ \ I_n]^T)^{-1} \cdot [A^T \ \ I_n], \tag{15}$$

$$s = [A^T \ \ I_n] \cdot ([A^T \ \ I_n] \cdot [A^T \ \ I_n]^T)^{-1} \cdot c. \tag{16}$$

Variables y and z need to be projected. New projection equations are computed from (4), (15) and (16).

$$y_p = y - A \cdot (A^T \cdot A + I_n)^{-1} \cdot (A^T \cdot y + z - c), \tag{17}$$

$$z_p = z - (A^T \cdot A + I_n)^{-1} \cdot (A^T \cdot y + z - c). \tag{18}$$

The matrix $(A^T \cdot A + I_n)^{-1}$ is computed before the iterative process, using the Sherman–Morrison–Woodbury expression [11], which provides a reduction in the number of floating-point operations because the dimension of the new matrix is smaller. Therefore,

$$(A^T \cdot A + I_n)^{-1} = I_n - A^T \cdot (A \cdot A^T + I_m)^{-1} \cdot A. \tag{19}$$

The Cholesky factorization of matrix $(A \cdot A^T + I_m)^{-1}$ is computed only once, before the iterative process.

5. Hopfield interior point approaches

Hopfield neural networks and interior point method are used to calculate an advanced initialization for infeasible primal–dual method.

Two alternatives of cooperation for warm start are defined: Hopfield primal–dual and Hopfield predictor–corrector. The new approaches use the improvement on Hopfield neural network discussed in the previous sections.

The Hopfield primal–dual and Hopfield predictor–corrector differ only in a direction used on step B.2 of warm start process. The general methodology used in these approaches is described below.

(A) *Feasibility process*: The feasibility process described in Section 4.2 gives initial primal–dual interior point.
(B) *Advanced initialization*: Enhanced starting points for the interior point optimization procedures are computed with a fixed number of the following cycle of three steps:
1. Compute a search direction, d^l, using either the classical primal–dual or the predictor–corrector interior point method;
2. Compute a new point along the search direction d^l, using a fixed step size (as previously discussed, we consider $\alpha^l = 1$);
3. The *feasibility process* is used again to convert the new point into an interior point.
(C) *Optimization*: Either the classical primal–dual or the predictor–corrector interior point method takes over the optimization process, beginning with the enhanced starting point.

6. Case studies

The approaches discussed in the previous sections were coded with MATLAB and applied to a subset of the *Netlib* collection of problems [13] (http://www.netlib.org). All problems were preprocessed [14,15] by the LIPSOL (Linear Programming Interior Point Solver v0.4) [16], (http://www.caam.rice.edu/∼zhang/lipsol/), before the computation. Case studies were run on a PC Pentium 4 1.8 GHz, with 500 MB of RAM, under Windows 2000.

Three parameters are used to compute the initial point by Hopfield-interior point methods:

- *Na*—the number of iterations of the Hopfield-interior-point-method to obtain enhanced starting points for interior point methods. $Na = 1$ was used;
- *Tol*—tolerance for convergence of the feasibility process. The methods presented in this work are infeasible interior point method that is why it is not necessary to use an initial feasible point. In this case the feasible process was applied only for $Tol = 100$ iterations.
- *It*—lower interior tolerance, used by the activation function ($g_{It}(x)$).

The case studies are organized in two sections. The next section presents the result of the classical primal–dual interior point when the warm start is applied. The predictor–corrector method is coming after.

6.1. Case studies for classical primal–dual interior point method

Table 1 presents result for the applications of the classical primal–dual methods. The first column presents the name and size of the problem; the second (P-D-C column) and third give the number of iterations and time to reach optimality using a classical initialization; the fourth (H-P-D column) and fifth give the number of iterations and time to reach optimality when advanced initialization by Hopfield neural network provides enhanced starting point (warm start) for the interior point method.

Table 1 shows that in all cases the number of iterations of the classical primal–dual method benefits from warm start provided by Hopfield interior point approaches but the time does not have the same gain except for one problem (STOCFOR2). In the implementation of the case studies was observed that the number of iterations is reduced when the initial point is somewhat centralized in the interior of the feasible region—this can be empirically achieved by controlling the lower tolerance of the activation function (*It*).

6.2. Case studies for predictor–corrector method

The performance of the predictor–corrector method with advanced initialization by Hopfield neural network is compared with the results obtained by the LIPSOL to solve the same problems.

Table 2 presents the results obtained by the two approaches. The first column presents the problem and its size. The second column presents the number of iterations of the predictor–corrector method with a classical initialization

Table 1
Results for the primal–dual interior point method

Problems ($N \times M$)	P-D-C (Iter)	Time (s)	H-P-D (Iter)	Time (s)
AFIRO (27 × 51)	11	0.08	8	0.31
SHARE1B (112 × 248)	32	0.22	29	0.89
SCTAP1 (300 × 660)	34	0.39	24	2.23
ISRAEL (174 × 316)	33	0.8	27	1.92
SCSD6 (147 × 1350)	15	0.19	11	2.79
SCFXM2 (644 × 1184)	41	1.16	34	5.14
SHIP12S (466 × 2293)	35	2.57	24	4.61
STOCFOR2 (2157 × 3054)	50	56.09	39	55.47
SCSD8 (397 × 2750)	17	0.33	12	8.79
SCTAP3 (1480 × 3340)	31	5.94	18	29.81
SHIP121 (838 × 5329)	38	28.91	29	50.80
TRUSS (1000 × 8806)	25	15.71	18	50.39
D2Q06C (2171 × 5831)	67	126.18	63	144.14
WOODW (1098 × 8418)	54	14.19	32	19.11

(LIPSOL). The third shows the number of iterations of the predictor–corrector to solve the problems using advanced initialization by neural networks.

In this case study it was not possible to compare the solution times. The methods were not codified in the same language; Hopfield predictor–corrector was codified in Matlab and LIPSOL has some routines programming in Fortran language.

The Table 2 shows that the predictor–corrector with warm start sometimes reaches the optimality after a smaller number of iterations than the predictor–corrector interior point method.

7. Conclusions

This paper discussed benefits from cross-fertilization between neural networks and optimization ideas to solve linear programming problems. A Hopfield neural network approach is used to go part of the way in the optimization process, where it passes the baton to primal–dual interior point methods for linear programming.

An alternative of cooperation between Hopfield networks and primal–dual methods was proposed: the classical primal–dual method or the predictor–corrector method and Hopfield networks going further in the optimization process, enabling advanced initializations for the interior point methods. Both alternatives were applied to a set of real-life linear programming problems from *Netlib* [13]—improvements resulting from the use of the Hopfield approach to linear optimization presented in the paper provided the basis for the great leap from solving illustrative problems to managing the real-life optimization problems provided by *Netlib*.

To sum up, the integrated approaches proposed here led to promising results, indicating that there may be a place for neural networks in real-life optimization problems. Further studies must be carried on to confirm these indications;

Table 2
Results for the predictor–corrector interior point method

Problems ($N \times M$)	LIPSOL (Iter)	Warm start (Iter)
Afiro (27 × 51)	6	8
Share1b (112 × 248)	22	19
Sctap1 (300 × 660)	12	17
Israel (174 × 316)	23	15
Scsd6 (147 × 1350)	11	8
Scfxm2 (644 × 1184)	21	20
Ship12s (466 × 2293)	14	18
Stocfor2 (2157 × 3054)	20	21
Scsd8 (397 × 2750)	7	11
Sctap3 (1480 × 3340)	16	18
Ship121 (838 × 5329)	16	18
Truss 1000 × 8806)	17	19
D2q06c (2171 × 5831)	28	32
Woodw (1098 × 8418)	26	28

however, there is enough evidences to expect that there are potential advantages in the frontier between the neural networks and optimization fields.

Acknowledgements

The authors are grateful for the support of "Fundação de Amparo à Pesquisa do Estado de São Paulo" (FAPESP) and "Conselho Nacional de Desenvolvimento Científico e Tecnológico" (CNPq).

References

[1] Wright SJ. primal–dual interior-point methods. Philadelphia, PA, USA: SIAM; 1996.
[2] Monteiro RDC, Ilan Adler, Resende MGC. A polynomial-time primal–dual affine scaling algorithm for linear and convex quadratic programming and its power series extension. Mathematics of Operations Research 1990;15:191–214.
[3] Mehrotra S. On the implementation of a primal–dual interior point method. SIAM Journal on Optimization 1992;2(4):575–601.
[4] da Silva IN. A neuro-fuzzy approach to systems optimization and robust estimation. PhD thesis, School of Electrical Engineering and Computer Science, State University of Campinas, Brazil; 1997 [in Portuguese].
[5] Yildirim EA, Wright SJ. Warm-start strategies in interior-point methods for linear programming. SIAM Journal on Optimization 2002;12(3): 782–810.
[6] Roos C, Terlaky T, Vial J-PH. Theory and algorithms for linear optimization. England: Wiley; 1997.
[7] Gondzio J. Warm start of the primal–dual method applied in the cutting plane scheme. Mathematical Programming 1998;83(1):125–43.
[8] Gondzio J, Vial J-P. Warm start and epsilon-subgradients in the cutting plane scheme for block-angular linear programs. Computational Optimization and Applications 1999;14:17–36.
[9] Luenberger D. Linear and nonlinear programming. Reading, MA: Addison-Wesley; 1984.
[10] Aiyer SVB, Niranja M, Fallside F. A theoretical investigation into the performance of the Hopfield model. IEEE Transactions on Neural Networks 1990;1(2):204–15.

[11] Golub GH, Van Loan CF. Matrix computations. 3rd ed, Baltimore: the Johns Hopkins University Press; 1996.
[12] da Silva IN, Ramos de Arruda LV , Wagner Caradori do Amaral. Nonlinear optimization using a modified Hopfield model. IEEE-World Congress on Computational Intelligence; 1998. p. 1629–33.
[13] Gay DM. Electronic mail distribution of linear programming test problems. Mathematical Programming Society COAL Newsletter 1985;13: 10–2.
[14] Andersen ED, Andersen KD. Pre-solving in linear programming. Mathematical Programming 1995;71:221–45.
[15] Gondzio J. Pre-solve analysis of linear programming prior to applying and interior point method. ORSA Journal on Computing 1997;9:73–91.
[16] Zhang Y. Solving large-scale linear programs by interior-point methods under the Matlab environment. Optimization Methods & Software 1998;10(1):1–31.