AN INTERIOR POINT METHOD FOR LINEAR PROGRAMMING USING WEIGHTED ANALYTIC CENTERS

JULIANNE ANDERSON, Department of Statistics and Operations Research, The University of North Carolina, Chapel Hill, NC 27599-3260; and
SHAFIU JIBRIN, Department of Mathematics and Statistics, Northern Arizona University, Flagstaff, AZ 86011-5717

ABSTRACT

Let R be the convex subset of $x \in \mathbb{R}^n$ defined by q linear inequalities $a_j^T x \leq b_j$, $j = 1,2,\ldots,q$ where x, $a_j \in \mathbb{R}^n$ and $b_j \in \mathbb{R}$. Given a strictly positive vector $\omega \in \mathbb{R}^q$, the *weighted analytic center* $x_{ac}(\omega)$ is the minimizer of the strictly convex function $\phi_\omega(x) = -\sum_{j=1}^{q} \omega_j \log(b_j - a_j^T x)$ over the interior of R. We consider the linear programming problem (LP): $\max\{c^T x \mid x \in R\}$. We give an interior point method for solving the LP that uses weighted analytic centers. We test its performance and limitations using a variety of LP problems. We also compare the method with the well-known logarithmic barrier method.

INTRODUCTION

The most common methods for linear programming move along the boundary of the feasible region to find the optimal solution. Most of these relate to the well-known simplex method. Another branch of methods are called interior point methods. These methods approach the optimal solution from the interior of the feasible region. We consider the linear programming problem (LP):

$$\text{Maximize } c^T x \tag{1.1}$$
$$\text{subject to } a_j^T x \leq b_j \ , j = 1 \ldots q \tag{1.2}$$

where a_j, $c \in \mathbb{R}^n$, $b_j \in \mathbb{R}$. This can also be written as:

$$\text{Maximize } c^T x$$
$$\text{subject to } Ax \leq b$$

where $A^T = [a_1, a_2, \ldots a_q]$, $b = [b_1, b_2, \ldots b_q]^T$. The dual is given by

$$\text{Minimize } b^T y$$
$$\text{subject to } A^T y = c,\ y \geq 0$$

where $y \in \mathbb{R}^q$. A point $x^* \in \mathbb{R}^n$ satisfies the Karush-Kuhn-Tucker (KKT) conditions if there exists $y^* \in \mathbb{R}^{q\,n}$ such that

1. $Ax^* \leq b$
2. $A^T y^* = c,\ y^* \geq 0$
3. $(y^*)^T (b - Ax^*) = 0$, i.e., $y_j^*(b_j - a_j^T x^*) = 0,\ j = 1,2,\ldots,q$

It is known that if (x^*, y^*) satisfies the KKT conditions, then x^* is primal optimal and y^* is dual optimal. If x is primal feasible and y is dual feasible, then the difference $b^T y - c^T x$ is called the **duality gap**. The duality gap is zero at optimality.

Most interior point methods use the idea of a primal-dual relationship to approach the optimal solution. These are categorized into three classes which are affine scaling methods, central trajectory methods, and potential reduction methods. Affine scaling methods create an ellipsoid within the feasible region that is centered at the current iterate. The ellipsoid is optimized to find a direction for the algorithm to proceed to the next iterate. This problem can be easily solved using the associated Karush-Kuhn-Tucker (KKT) system since it is a system of linear equations. Central trajectory methods approach the optimal solution through a central path. Algorithms that use this method are called path-following algorithms. The path is obtained as the optimal solution of a parameterized barrier function. Potential reduction methods use a potential function to find the next iterate with a better objective function value while still remaining in the interior of the feasible region (Freund and Mizuno 2000).

Each of the interior point methods described has its benefits. Affine scaling methods follow a comparably simple algorithm (see Sun et al. 1995 for an example). Alternately, central trajectory methods are used most often in practice, and they are most useful in theory (see Potra and Sheng 1995 for an example). Potential reduction methods are excellent because of their performance guarantee and because they can output dual information which allows the user to choose a tolerance level (see Faybusovich 1998 for an example).

While the idea of beginning in the interior of the feasible region was thought of as early as 1960, the

ANDERSON, J., AND S. JIBRIN. 2009. AN INTERIOR POINT METHOD FOR LINEAR PROGRAMMING USING WEIGHTED ANALYTIC CENTERS. *JOURNAL OF THE ARIZONA-NEVADA ACADEMY OF SCIENCE* 41(1):1-7.

first person to create an interior point method that could theoretically be better than the simplex method was Narendra Karmarkar of AT&T Bell Laboratories in 1984. For more examples of interior point methods, see <http://www.optimization-online.org/> and <http://www-unix.mcs.anl.gov/otc/InteriorPoint>. This paper presents a potential reduction interior point method using weighted analytic centers. This is similar to the method presented in Reneger (1988). The basic idea of this method is to decrease the size of the feasible region, R, at each iteration until the weighted analytic center, an interior point, is close to the optimal solution. We also compare this method with the logarithmic barrier method.

WEIGHTED ANALYTIC CENTER

In this section, we introduce the concept of a weighted analytic center and discuss some of its properties.

Let R be the convex subset of IR^n defined by the q linear inequality constraints (1.2). We assume R is bounded and full dimensional. Given a strictly positive weight vector $\omega = (\omega_1, \omega_2, \ldots \omega_q)^T$, the *weighted analytic center* $x_{ac}(\omega)$ is the minimizer of the strictly convex barrier function

$$\phi_\omega(x) = -\sum_{j=1}^{q} \omega_j \log(b_j - a_j^T x)$$

over the interior of R (Atkinson and Vaidya 1992, Jibrin and Swift 2004). An example of $\varphi_\omega(x)$ is given in Figure 1. If $\omega = [1,1,\ldots,1]^T$, $x_{ac}(\omega)$ is called the *analytic center*. Intuitively, the analytic center of the constraints set (1.2) is the point in the interior of R that R *balances* the distances to each of the constraints. The weighted analytic center is the analytic center of the constraints with weights given to each constraint. Each weight pushes the analytic center away from the corresponding constraint. A Newton method for computing $x_{ac}(\omega)$ is given by

$$x_{k+1} = x_k - H_{\phi_\omega}(x_k)^{-1} \nabla \phi_\omega(x_k)$$

where $\nabla\varphi_\omega(x_k)$ is the gradient and $H_{\phi_\omega}(x_k)$ is the Hessian of φ_ω at x_k and

$$\nabla\phi_\omega(x) = \sum_{j=1}^{q} \omega_j \frac{a_j}{b_j - a_j^T x}$$

$$H_{\phi_\omega}(x) = \sum_{j=1}^{q} \omega_j \frac{a_j a_j^T}{(b_j - a_j^T x)^2}$$

We will consider the following LP throughout this paper. The analytic center of this LP is $x^*=(2.1914, 1.7400)$ as shown in Figure 2.

LP Example: Maximize $Z=3x_1+4x_2$

$$\begin{aligned} \text{subject to } x_1+2x_2 &\le 10 \quad (1)\\ 2x_1+x_2 &\le 15 \quad (2)\\ x_1 &\ge 0 \quad (3)\\ x_2 &\ge 0 \quad (4) \end{aligned}$$

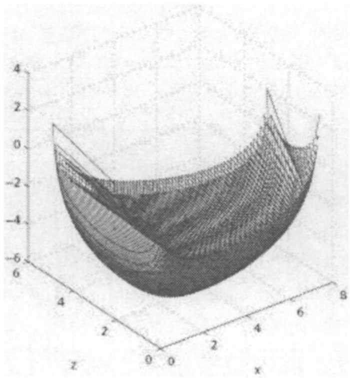

Figure 1. *The barrier function $\varphi_\omega(x)$ for the example. It shows $\varphi_\omega(x)$ is strictly convex over R.*

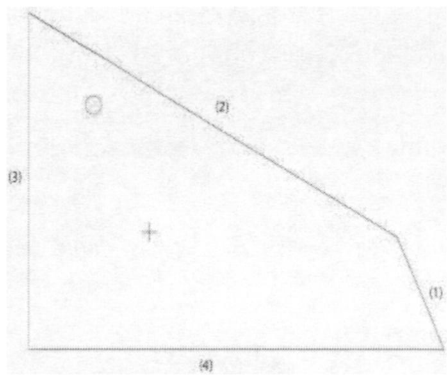

Figure 2. *The central + is the analytic center of the LP. The open circle ○ is the weighted analytic center with a weight of 5 on the constraint (4), $\omega=[1,1,1,5]^T$.*

Theorem 2.1 (Jibrin and Swift 2004)

Every point in the interior of the feasible region R is a weighted analytic center.

Proof. Let x_0 be an interior point of R. Then $b_j - a_j^T x_o > 0$ for all j. Then x_o is a weighted analytic center if and only if for all j there exists $\omega_j > 0$ that satisfy

$$\sum_{j=1}^{q}\omega_j \frac{a^j}{b_j - a_j^T x^*} = 0.$$

Let x^* be the analytic center of R. Then, $\omega=[1,1,1,1]^T$

$$\nabla \varphi_\omega\left(x^*\right) = \sum_{j=1}^{q} \frac{a^j}{b_j - a_j^T x^*} = 0.$$

Since x^* is an interior point of R, then $b_j - a_j^T x^* > 0$ for all j. Let

$$\omega_j = \frac{b_j - a_j^T x_0}{b_j - a_j^T x^*}, \text{ for } j, 1,2,..,q$$

Then

$$\sum_{j=1}^{q}\omega_j \frac{a^j}{b_j - a_j^T x_0} = \sum_{j=1}^{q} \frac{b_j - a_j^T x_0}{b_j - a_j^T x^*} \cdot \frac{a^j}{b_j - a_j^T x_0} = \sum_{j=1}^{q} \frac{a^j}{b_j - a_j^T x^*} = 0$$

And so the weighted analytic center using the weights ω_j is x_0.

Now suppose the desired weighted analytic center is $x_0=(1,1)^T$. What weights would give us this value? Using the results above, the necessary weights are approximately: $\omega=[1.6172,1.3518, 0.4563,0.5747]^T$.

REPELLING LIMITS

This section discusses repelling limits for linear inequality constraints and errors that happen when finding these limits. We show that the repelling limit is a boundary point of the feasible region.

Consider the positive weight vector $\omega \in \mathrm{IR}^q$ with $\omega_k=\mu$ and $\omega_j=1$ for all $j \neq k$. A repelling limit for the k^{th} constraint is defined as the $\lim_{\mu\to\infty} x^{(k)}(\omega)$ (Caron et al. 2002, Jibrin and Swift 2004) where We give the following result.

$$x^{(k)}(\omega) = argmin\left\{ \mu \log(b_k - a_k^T x) + \sum_{j=1, j\neq k}^{q} \log(b_j - a_j^T x) \mid x \in \mathrm{R} \right\}.$$

Theorem 3.1

A repelling limit $\lim_{\mu\to\infty} x^{(k)}(\omega)$ for the k^{th} constraint is a boundary point of the feasible region R defined by the system (1.2).

Proof. According to Lemma 3.1 in Caron et al. (2002), the slack s_k associated with the k^{th} constraint is bounded away from 0. The variable s_k is maximized over as $\mu\to\infty$. Hence, $\lim_{\mu\to\infty} x^{(k)}(\omega)$ is a boundary point of R.

Theorem 3.1 shows that by increasing the weight μ to a very high value we can approach the boundary of the region R as close as we may want from the interior. However, due to numerical errors from Newton method and the difficulty of choosing the correct weight that is large enough, the boundary may never be reached as desired.

Figure 3 shows the path that the weighted analytic center follows when the weight on one constraint increases from 1-5 with all weights on the other constraints held constant at 1. Figure 4 shows the path as well, but with the weight on one constraint increasing from 1-30. As the weight increases, Newton's method can no longer compute the weighted analytic center. The corresponding weighted analytic centers of those constraints which do not end in a star output errors when computing

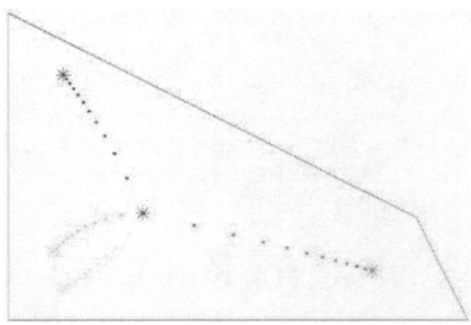

Figure 3. *Path of weighted analytic center as the weight of a constraint increases from 1-5 while the others weights are fixed at 1.*

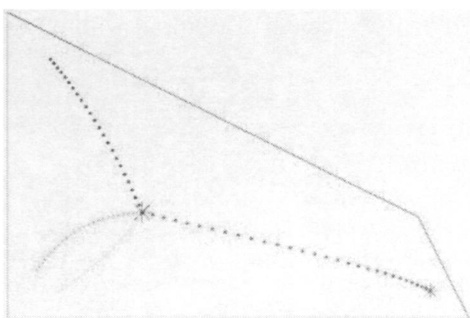

Figure 4. *Path of weighted analytic center as the weight of a constraint increases from 1-30 while the others weights are fixed at 1. Note that the paths did not reach the boundary of the feasible region.*

$$\sum_{j=1}^{q}\omega_j \frac{a_i^j}{b_j - a_j^T x_0} = 0,$$

Newton's method and thus could not calculate the final results with such large weights. The one corresponding constraint with the star was the only of the four constraints that could find the weighted analytic center for all of the given weights. Meaning, Newton's method was able to compute each new weighted analytic center when increasing the weight from 1 to 30 on this constraint.

From these figures it is clear that due to errors, the weighted analytic centers did not approach the boundary of R. However, from Theorem 3.1, we suspect that each of corresponding repelling limits would approach a corner point of R. As a result of numerical problem with Newton's method when large weight is used, the algorithm we present will use a moderate weight that Newton's method can handle while still significantly decreasing the size of the region R.

Now, supposing that there were no numerical errors, how large would this weight need to be to be at optimality? This depends upon the LP and how close one needs to be to be considered optimal. It is difficult to find this value.

AN ALGORITHM FOR SOLVING THE LP

This section presents our approach to solving an LP from the interior of the region using a sequence of weighted analytic centers.

First, the algorithm creates a new constraint through the analytic center of the LP whose boundary is normal to the objective function vector c. The weighted analytic center of this new set of constraints is found with a weight added to the new constraint. This process is repeated until the weighted analytic center is optimal within a given tolerance.

Algorithm

Initialization:

Choose an interior point x_0 of R.
Choose a weight $\omega = [1,\ldots 1,8]$ and tolerance levels, $tol1$ and $tol2$.
Set iter= 1
Find the analytic center x_{ac} of the system $Ax \le b$ starting from x_0.
Set $x^* = x_{ac}$
Set $A_{new}=[A;-c^T]$
Set $b_{new}=[b;-c^T(x_{ac}-(tol2)c)]$
Find the weighted analytic center($x_{ac}(\omega)$) of $A_{new}x \le b_{new}$ starting from x^*.
while $\|x^*-x_{ac}(\omega)\|>tol1$
  Set iter = iter+1
  Set $x^*=x_{ac}(\omega)$
  Set $b_{new}=[b;-c^T(x_{ac}-(tol2)c)]$
  Find the weighted analytic center $x_{ac}(\omega)$ of $A_{new}x \le b_{new}$ starting from x^*.
end

Figure 5. Shows the new constraint created at each iteration.

In the algorithm, we insert a new constraint into the system that is normal to the objective function vector (c) and whose boundary passes through the analytic center as shown in Figure 5. We move this constraint a distance of $tol2$ in the opposite direction of c. We use the current weighted analytic center as a starting point for computing the new weighted analytic of the new system of constraints. However, if the weighted analytic center is too close to the boundary of the new constraint, errors with Newton's may appear which is the reason for moving the new constraint away from the weighted analytic center, and so $tol2$ should be carefully selected. Figures 6 and 7 show sequences of iterates of the algorithm.

NUMERICAL EXPERIMENTS AND COMPARISON WITH THE LOGARITHMIC BARRIER METHOD

In this section, we use numerical experiments to investigate the effectiveness of our method and compare it with the well-known logarithmic barrier method.

We ran our algorithm using a tolerance 1 ($tol1$) level of 0.00000008, a tolerance 2 ($tol2$) level (the amount to push back the new constraint) of 0.0001, and a weight of 8, on the new constraint, that is, μ=8 and ω=[1,...,1,8]. We also ran the logarithmic barrier method using a tolerance level of 0.00000008, an initial weight μ_0 of 0.5, and use a factor of 1.2 to increase μ_k.

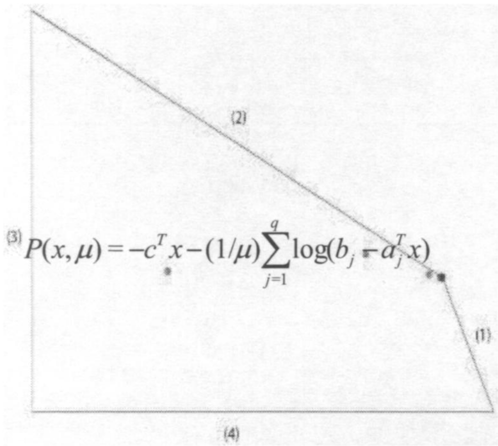

Figure 6. *The sequence of iterates of the algorithm using ω=[1,1,1,8].*

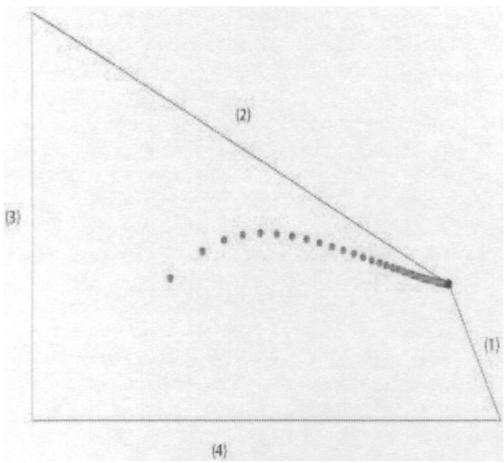

Figure 7. *The sequence of iterates of the algorithm using ω=[1,1,1,1.2].*

Similar to our algorithm, the logarithmic barrier method uses the idea of a weight. However, because the weight increases at each iteration, it is often difficult to solve the LP using Newton's method in the logarithmic barrier method. A description of the logarithmic barrier method applied to our original LP (1.1)-(1.2) is as follows. Let

Logarithmic Barrier Algorithm

Initialization:

Choose an initial interior point x_s^0 of R.

Choose an initial barrier parameter value, μ_0=0.5, and choose a tolerance level, say 0.00000008.

Set k = 0.

while $q\frac{1}{\mu_k} > tol$ (duality gap)

Find x_k=argmin$\{P(x,\mu_k)|x\in \mathrm{IR}^n\}$ starting with x_s^0 (Newton's Method)

Choose a new starting point by setting $x_s^{k+1}=x_k$

Choose a new weight, $\mu_{k+1}=1.2\mu_k$.

Set k=k+1.

end

The duality gap for the original problem is $q\frac{1}{\mu_k}$ (Boyd and Vanderberghe 2008). This is the difference between the objective function values of the primal and the dual at the current iteration k. This gap is 0 at optimality. Using this method, one should use small weight, say μ_0=0.5. If Newton's method turns out to be quick, the weights can be increased more quickly. Figure 8 shows a sequence of iterates for the logarithmic barrier method.

The test problems used were randomly generated, and the exact optimal solution of each is given found using a standard LP solver. The results are given in Table 1.

The values recorded include the number of constraints, the number of variables, the exact optimal solution, the optimal solution found using the algorithm, the number of iterations our algorithm used, the result using the logarithmic barrier method, and the number of iterations the logarithmic barrier method used.

The algorithm we present always finds the optimal solution, however, its accuracy is less than perfect. On the other hand, the logarithmic barrier method has pin-point accuracy. The better accuracy may be due to its stopping criterion. It uses the duality gap as a stopping criteria while our algorithms stops when the iterates become very close. However, logarithmic barrier method uses a lot more iterations than our method.

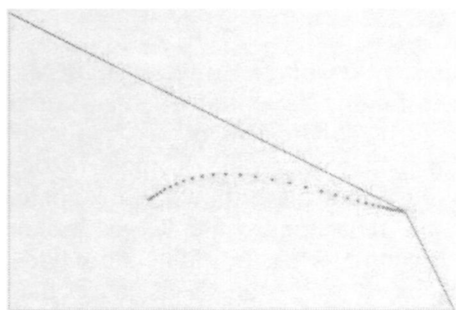

Figure 8. *The sequence of iterates of the log barrier algorithm.*

Table 1. *Test problems and numerical results.*

Problem	No. of constraints	No. of variables	Actual optimal solution	Optimal solution using algorithm	No. of interations	Log barrier	No. of interations
Main Ex.	4	2	$\begin{bmatrix} 6.666667 \\ 1.666667 \end{bmatrix}$	$\begin{bmatrix} 6.666542 \\ 1.666682 \end{bmatrix}$	14	$\begin{bmatrix} 6.666667 \\ 1.666667 \end{bmatrix}$	114
LP1	4	2	$\begin{bmatrix} 1.884216 \\ 0.1633870 \end{bmatrix}$	$\begin{bmatrix} 1.884114 \\ 0.1633686 \end{bmatrix}$	13	$\begin{bmatrix} 1.884216 \\ 0.1633870 \end{bmatrix}$	114
LP2	10	3	$\begin{bmatrix} 10.18029 \\ -6.566789 \\ -2.108501 \end{bmatrix}$	$\begin{bmatrix} 10.18006 \\ -6.566731 \\ -2.108453 \end{bmatrix}$	10	$\begin{bmatrix} 10.18029 \\ -6.566789 \\ -2.108501 \end{bmatrix}$	119
LP3	30	5	$\begin{bmatrix} 2.3442540 \\ -0.73313825 \\ 0.80580871 \\ -2.1052295 \\ -0.30585000 \end{bmatrix}$	$\begin{bmatrix} 2.3441769 \\ -0.73313827 \\ 0.80573610 \\ -2.1051971 \\ -0.30588869 \end{bmatrix}$	18214	$\begin{bmatrix} 2.344254 \\ -0.7331382 \\ 0.8058087 \\ -2.105229 \\ -0.3058500 \end{bmatrix}$	125
LP4	20	4	$\begin{bmatrix} -0.4086974 \\ 0.1926800 \\ 0.04006089 \\ 1.528117 \end{bmatrix}$	$\begin{bmatrix} -04086872 \\ 0.1926097 \\ 0.04002882 \\ 1.58030 \end{bmatrix}$	24	$\begin{bmatrix} -0.4086974 \\ 0.1926800 \\ 0.04006089 \\ 1.528117 \end{bmatrix}$	123
LP5	30	7	$\begin{bmatrix} -3.167578 \\ 1.182596 \\ -3.068419 \\ 1.049359 \\ -4.667124 \\ 1.771037 \\ -3.264342 \end{bmatrix}$	$\begin{bmatrix} -3.167394 \\ 1.182405 \\ -3.068407 \\ 1.048910 \\ -4.667005 \\ 1.770600 \\ -3.264620 \end{bmatrix}$	30	$\begin{bmatrix} -3.167578 \\ 1.182596 \\ -3.068419 \\ 1.049359 \\ -4.667124 \\ 1.771037 \\ -3.264342 \end{bmatrix}$	125
LP6	40	3	$\begin{bmatrix} -0.1380519 \\ -0.3359732 \\ -1.313663 \end{bmatrix}$	$\begin{bmatrix} -0.1380986 \\ 0.4474045 \\ -1.313796 \end{bmatrix}$	18	$\begin{bmatrix} -0.1380519 \\ -0.3359732 \\ -1.313663 \end{bmatrix}$	127
LP7	100	3	$\begin{bmatrix} 0.3514976 \\ 1.332212 \\ -0.1231155 \end{bmatrix}$	$\begin{bmatrix} 0.3514534 \\ 1.332139 \\ -0.1231137 \end{bmatrix}$	17	$\begin{bmatrix} 0.3514976 \\ 1.332212 \\ -0.1231155 \end{bmatrix}$	132

CONCLUSION

We present an initial investigation of one approach to finding the optimal solution of an LP using weighted analytic centers.

The method works well on most of our test problems, but not on all. Still, it has great potential. For example, it uses close iterates as a stopping criterion. Meaning, when the old iterate and the new iterate become close, it stops. This causes the optimal solution to have limited accuracy. One idea to improve this issue is to use the duality gap as a stopping criterion. That is, stop when the duality gap is close to 0. This is how the logarithmic barrier finds its stopping point, and it is far more accurate.

Another idea to improve this method is to change the starting point. The algorithm currently begins at the analytic center of the LP. Because of this, the algorithm only works with completely bounded regions. If instead we begin with the initial starting point x_0, we could remove the bounded restriction on many LP problems. Since our algorithm creates a new constraint through the first point, this new constraint would bound an unbounded feasible region.

There is also a weakness in the algorithm in that the new constraint must be moved back. This constraint moves back a fixed amount each iteration. The reason for this is that the new constraint is placed through the weighted analytic center which is used as the x_0 for finding the new weighted analytic center. If x_0 is not in the interior of the region, this cannot be computed. Additionally, if x_0 is too close to the boundary of the region, the computer cannot handle the computations necessary to find the new weighted analytic center. For this reason, we chose to push the new constraint back 0.0001. However, it seems reasonable that with systems with larger areas, the amount to push back the constraint would need to be larger than with systems with smaller areas. Part of the reason the algorithm is not as accurate as it could be is because of this constraint being pushed back. If we keep pushing back the constraint when the iterations become close together, we are losing an increasing amount of the progress we made each iteration. One idea to fix this problem is to use a variable amount to push the constraint back based on the size of the system or the progress from iteration to iteration. Even better would be a method that allows us to find the weighted analytic center without having to push the constraint back at all.

When finding the weighted analytic center $x_{ac}(\omega)$ using a large weight on one of the weights, Newton's method runs into numerical errors. One suggestion is to use a different method. For example, Atkinson and Vaidya (1992) introduce a method that avoids this problem.

Our method uses simple thought out process to finding the optimal solution. It uses an idea that many beginning operations research courses teach to graphically find the optimal solution of an LP. The algorithm is not overly complicated, and the results are very good. Despite its limitations, the method generally approaches the optimal solution and sometimes quickly, and so overall the method is a success.

ACKNOWLEDGMENTS

This work was initially done as an undergraduate research experience by the first author at Northern Arizona University under the supervision of the second author in the spring of 2008.

REFERENCES

ATKINSON, D. S., and P. M. VAIDYA. 1992. A scaling technique for finding the weighted analytic center of a polytope. *Math. Prog.* 57:163-192.

BOYD, S., and L. VANDENBERGHE. 2008. *Convex Optimization.* Cambridge University Press, New York, NY.

CARON, R. J., H. J. GREENBERG and A. G. HOLDER. 2002. Analytic centers and repelling inequalities. *European Journal of Operational Research* 143(2):268-290.

FAYBUSOVICH, L. 1998. *A Jordan-Algebraic Approach to Potential-Reduction Algorithms.* Research report, Department of Mathematics, University of Notre Dame.

FREUND, R., and S. MIZUNO. 2000. Interior point methods: Current status and future direction. Pp. 441-466 in *High Performance Optimization*. Kluwer Academic Publishers.

OPTIMIZATION ONLINE. <http://www.optimization-online.org/> Accessed 3/12/09.

INTERIOR-POINT METHODS ONLINE. <http://www-unix.mcs.anl.gov/otc/InteriorPoint> Accessed 3/12/09, directs you to Optimization Online.

JIBRIN, S., and J. SWIFT. 2004. The boundary of weighted analytic centers for linear matrix inequalities. *Journal of Inequalities in Pure and Applied Mathematics*, 5(1):art. 14. Online at <http://www.univie.ac.at/EMIS/journals/JIPAM/images/158_03_JIPAM/158_03_www.pdf>

POTRA, F. A., and R. SHENG. 1995. A path following method for LCP with superlinearly convergent iteration sequence. *Reports on Computational Mathematics*, No. 69. Department of Mathematics, University of Iowa.

RENEGER, J. 1988. A polynomial-time algorithm, based on Newton's method, for linear programming. *Mathematical Programming* 40:59-93.

SUN, J., J. ZHU, and Z. GONG. 1995. A primal-dual affine scaling algorithm with necessary centering as a safeguard. *Optimization* 35:333-343.