 ISSN 1984-820X

Detecção de Linhas Redundantes em Problemas de Programação Linear de Grande Porte

Aurelio R. L. de Oliveira , **Daniele Costa Silva**,
Depto de Matemática Aplicada, IMECC, UNICAMP,
13083-859, Campinas, SP
E-mail: aurelio@ime.unicamp.br, silvadc@yahoo.com.br,

RESUMO

A presença de linhas redundantes na matriz de restrições não é incomum em problemas reais de grande porte. A existência de tais linhas deve ser levada em consideração na solução destes problemas. Principalmente nos casos em que a única forma possível de solução é resolver os sistemas lineares oriundos dos métodos de pontos interiores por métodos iterativos. Nesta situação as linhas redundantes representam uma dificuldade considerável, pois geram uma matriz singular e os métodos iterativos não convergem. A única alternativa viável consiste em detectar tais linhas e eliminá-las antes da aplicação do método. A seguir apresentaremos uma técnica sofisticada baseada na decomposição LU de uma base e sua atualização para encontrar todas as linhas redundantes de uma matriz de restrições. Tomando como base as idéias propostas em [1].

1. O algoritmo

Considere sem perda de generalidade o seguinte problema de programação linear:

$$
\begin{aligned}
&\text{Maximizar} \quad \sum_{j=1}^{n} c_j x_j \\
&\text{sujeito} \quad \text{a} \quad \sum_{j=1}^{n} a_{ij}x_j + x_{n+i} = b_i \quad (i = 1, 2, ..., m) \\
&\qquad l_j \le x_j \le u_j \quad (j = 1, 2, ..., n) \\
&\qquad 0 \le x_{n+i} \le 0 \quad (i = 1, 2, ..., m).
\end{aligned}
\tag{1}
$$

Onde $A = [a_{ij}] \in \mathbb{R}^{mxn}$; $x, l, u \in \mathrm{R}^n$, $b \in \mathrm{R}^m$ e $x_{n+1}, x_{n+2}, ..., x_{n+m}$ variáveis artificiais. Sendo assim, podemos apresentar o algoritmo que consiste em detectar as linhas redundantes do problema acima:

Passo 0: Seja S o conjunto de índices i tal que a variável artificial x_{n+i} é uma variável básica.
Passo 1: Se S é vazio pare. Caso contrário escolha $k \in S$.
Passo 2: Resolva o sistema $B^t r = e_k$, onde B é a matriz base e e_k representa o vetor canônico na posição k. Se $r^t A = 0$, podemos afirmar que a k-ésima linha de A é redundante. Caso contrário, há uma variável não básica $x_j \quad : \quad r^t c \neq 0$ para a correspondente coluna c de A. Substitua a k-ésima coluna de B por c e substitua x_{n+k} por x_j na base e retorne para 1. O algoritmo deve ser aplicado para todo $k \in S$.

2. A proposta de implementação

O algoritmo apresentado acima é bastante conhecido (ver [2]). Porém, sua implementação direta pode ter um custo computacional muito elevado. A seguir, propomos técnicas (heurística e decomposição LU) cujos objetivos são manter o menor número possível de variáveis artificiais na base, uma vez que o custo computacional do algoritmo está relacionado a esta quantidade, e tornar a solução do sistema $B^t r = e$ mais eficaz.

2.1. Heurística

Para manter o menor número possível de variáveis artificiais na base utilizaremos a heurística descrita a seguir.

Dizemos que uma linha é não atribuída se esta não está associada a alguma variável. Primeiro contamos o número de elementos não nulos em cada linha e em cada coluna. Associamos uma variável artificial à linha que contém o maior número de elementos não nulos. Removemos esta linha temporariamente do problema e atualizamos a contagem de elementos não nulos em cada coluna. Se alguma nova coluna única aparecer, associamos uma variável estrutural à linha em que ela ocorre. Desta forma criamos novas colunas únicas. Repetimos este processo até que cada linha esteja associada a uma variável. Através desta construção obtemos uma base inicial triangular superior e não singular.

2.2. Decomposição LU

O algoritmo consiste na resolução do sistema $B^t r = e$. A proposta é resolver este sistema por meio de decomposição LU, utlizando algumas técnicas para torná-la mais eficiente do ponto de vista computacional.

Assuma que exista k colunas artificiais na base, então reordenamos as linhas e colunas de B^t de forma que as últimas k linhas de B^t sejam iguais as últimas k linhas da matriz identidade.

Assim, o sistema $B^t r = e$, pode ser resolvido da seguinte forma:

$$Ls = e. \tag{2}$$

$$Ur = s. \tag{3}$$

Após a resolução dos sistemas podem ocorrer duas situações:

1. A coluna artificial se mantém na base, neste caso a redundância é detectada.
2. A coluna artiticial é substituída por uma coluna estrutural.

Na segunda situação a decomposição LU precisa ser atualizada. Neste caso, (agora assumindo que temos uma base inicial B de dimensão m e k colunas artificiais na base) substituimos a (m-k)-ésima linha de U por $L^{-t} a_j$ onde j é o índice da coluna entrante e armazenamos as operações que restauram a forma triangular de U como um produto de matrizes elementares, ou seja, armazenamos L^{-1} na forma produto [3].

3. Conclusões e trabalhos futuros

A detecção de linhas redundantes é uma ferramenta poderosa para resolução de problemas de grande porte de forma eficaz, pois atua nos fatores dimensão e redundância melhorando a estabilidade numérica de problemas que possuem uma grande quantidade de linhas redundantes. E em alguns casos é imprescindível no processo de resolução dos problemas (casos em que só é possível resolver o problema por métodos iterativos). Além disso, esta ferramenta pode ser utilizada em diversos contextos, o que reforça a importância da implementação de métodos eficazes para detecção de linhas redundantes como o aqui proposto.

Como trabalhos futuros pretendemos finalizar as implementações e expor os resultados computacionais.

Palavras-chave: *Linhas Redundantes. Método de Pontos Interiores. Decomposição LU.*

Referências

[1] E. D. Andersen, Finding all linearly dependent rows in large-scale linear programming, *Optimization Methods and Softwares*, 6 (1995) 219-227.

[2] V. Chvàtal, "Linear Programming", W. H. Freeman and Company, New York, 1983.

[3] I. Maros, "Computational Techniques of the Simplex Method", Kluwer, Boston, 2003.