

Implementação do tableau completo

Marina Andretta

ICMC-USP

14 de novembro de 2018

Baseado no livro Introduction to Linear Optimization, de D. Bertsimas e J. N. Tsitsiklis.

Implementação do tableau completo

Vamos ver agora a implementação do Método Simplex chamada de **tableau completo**.

Aqui, em vez de manter e atualizar a matriz B^{-1} , mantemos e atualizamos a matriz

$$B^{-1} (b \mid A)$$

com colunas $B^{-1}b$ e $B^{-1}A_1, \dots, B^{-1}A_n$.

Esta matriz é chamada de **simplex tableau**.

Implementação do tableau completo

Note que a coluna $B^{-1}b$, chamada de **coluna 0**, contém o valor das variáveis básicas. A coluna $B^{-1}A_j$ é chamada de i -ésima coluna do tableau.

A coluna $u = B^{-1}A_j$, que corresponde à variável que entra na base, é chamada de **coluna pivô**.

Se a ℓ -ésima variável básica sai da base, a ℓ -ésima linha do tableau é chamada de **linha pivô**.

O elemento da coluna e linha pivô é chamado de **elemento pivô**. Note que o elemento pivô é u_ℓ e é sempre positivo (a menos que $u \leq 0$, caso em que o critério de otimalidade é satisfeito e o algoritmo pára.)

Implementação do tableau completo

Uma interpretação para os elementos do tableau é a seguinte. As restrições de igualdade são dadas inicialmente na forma $b = Ax$.

Dada a base atual B , estas restrições de igualdade podem ser expressas, de maneira equivalente, como

$$B^{-1}b = B^{-1}Ax,$$

que é exatamente a informação armazenada no tableau.

Em outras palavras, as linhas do tableau fornecem os coeficientes das restrições de igualdade $B^{-1}b = B^{-1}Ax$.

Implementação do tableau completo

Ao final de cada iteração, precisamos atualizar o tableau $B^{-1}(b \mid A)$ e calcular $\bar{B}^{-1}(b \mid A)$.

Isso pode ser feito multiplicando o tableau pela esquerda pela matriz Q tal que $QB^{-1} = \bar{B}^{-1}$.

Como visto anteriormente, isso é o mesmo que realizar operações elementares em linhas de forma a transformar B^{-1} em \bar{B}^{-1} .

Para isto, somamos a cada linha um múltiplo da linha pivô de forma a transformar todas as entradas na coluna pivô em 0, com exceção do elemento pivô, que é transformado em 1.

Implementação do tableau completo

Para determinar a coluna $A_{B(\ell)}$ que sai da base e o tamanho de passo θ^* , fazemos o seguinte: $x_{B(i)}/u_i$ é a razão entre a i -ésima entrada da coluna 0 do tableau e a i -ésima componente na coluna pivô.

Consideramos somente índices i para os quais u_i é positivo.

A menor destas razões é θ^* e determina ℓ .

Implementação do tableau completo

É comum acrescentar ao tableau uma linha no topo, chamada de **linha 0**.

O elemento no canto superior esquerdo é o valor $-c_B^T x_B$, que é o negativo do custo atual.

O restante da linha 0 é o vetor de custos reduzidos, ou seja,
$$\bar{c}^T = c^T - c_B^T B^{-1} A.$$

Implementação do tableau completo

Então, a estrutura do tableau é dada por

$-c_B^T B^{-1}b$	$c^T - c_B^T B^{-1}A$
$B^{-1}b$	$B^{-1}A$

ou, em mais detalhes,

$-c_B^T x_B$	\bar{c}_1	...	\bar{c}_n
$x_{B(1)}$			
\vdots	$B^{-1}A_1$...	$B^{-1}A_n$
$x_{B(m)}$			

Implementação do tableau completo

A regra para atualizar a linha 0 é idêntica à usada para atualizar as demais linhas do tableau: somar um múltiplo da linha pivô à linha 0 de forma a transformar em 0 o custo reduzido da variável que entra na base.

Vejamos porque esta atualização funciona.

Implementação do tableau completo

No início de uma iteração, a linha 0 tem a forma

$$(0 \mid c^T) - g^T(b \mid A),$$

com $g^T = c_B^T B^{-1}$.

Portanto, a linha 0 é igual a $(0 \mid c^T)$ mais uma combinação linear das linhas de $(b \mid A)$.

Seja j a coluna pivô e ℓ a linha pivô. Note que a linha pivô tem a forma $h^T(b \mid A)$, com h^T a ℓ -ésima linha de B^{-1} .

Implementação do tableau completo

Portanto, depois que um múltiplo da linha pivô é somado à linha 0, esta linha é novamente igual a $(0 \mid c^T)$ mais uma outra combinação linear das linhas de $(b \mid A)$, e tem a forma

$$(0 \mid c^T) - p^T(b \mid A),$$

para um vetor p .

Lembre-se que a regra de atualização proposta é tal que o elemento da coluna pivô na linha 0 se torna 0, ou seja,

$$c_{\bar{B}(\ell)} - p^T A_{\bar{B}(\ell)} = c_j - p^T A_j = 0.$$

Implementação do tableau completo

Considere agora a coluna $\bar{B}(i)$, para $i \neq \ell$ (ou seja, uma coluna correspondente a uma variável básica que permanece na base).

O 0-ésimo elemento desta coluna é 0 antes da mudança de base, já que este é o custo reduzido de uma variável básica.

Como $B^{-1}A_{B(i)}$ é a i -ésima coluna da identidade e $i \neq \ell$, o elemento na linha pivô para esta coluna também é 0.

Portanto, somar um múltiplo da linha pivô à linha 0 não afeta esta coluna da linha 0.

Implementação do tableau completo

Concluimos, então, que o vetor p satisfaz $c_{\bar{B}(i)} - p^T A_{\bar{B}(i)} = 0$ para toda coluna $A_{\bar{B}(i)}$ na nova base.

Isso implica que $c_{\bar{B}} - p^T \bar{B} = 0$ e $p^T = c_{\bar{B}}^T \bar{B}^{-1}$.

Portanto, usando a regra de atualização proposta, a linha 0 do tableau atualizado é

$$(0 \mid c^T) - c_{\bar{B}}^T \bar{B}^{-1} (b \mid A).$$

Uma iteração do tableau completo

Descrevemos aqui uma iteração da implementação do **tableau completo**.

- P1. Em uma iteração típica, começamos com o tableau associado à uma base B e a solução básica viável correspondente x .
- P2. Examine os custos reduzidos na linha 0 do tableau.

Se todos os custos reduzidos forem não-negativos, a solução básica viável correspondente é ótima e o algoritmo pára.

Caso contrário, escolha um j para o qual $\bar{c}_j < 0$.

Uma iteração do tableau completo

P3. Considere o vetor $u = B^{-1}A_j$, que é a j -ésima coluna do tableau (coluna pivô).

Se nenhuma componente de u for positiva, temos $\theta^* = \infty$, custo ótimo é $-\infty$ e o algoritmo pára.

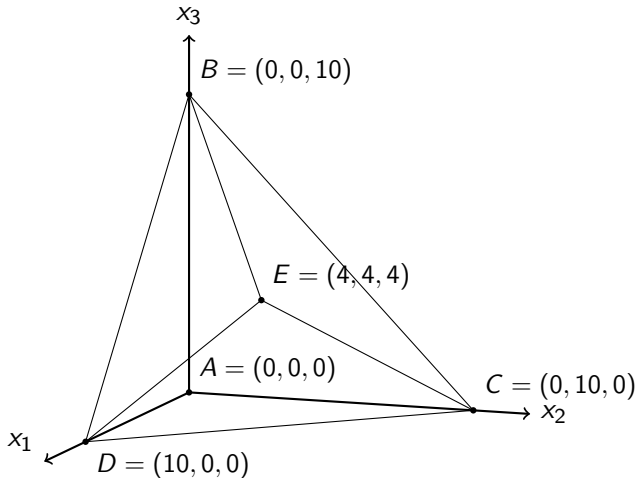
P4. Para cada componente u_i positiva, calcule as razões $x_{B(i)}/u_i$.

Seja ℓ o índice de uma linha que corresponde à menor razão. A coluna $A_{B(\ell)}$ sai da base e a coluna A_j entra na base.

P5. Some a cada linha do tableau um múltiplo da ℓ -ésima linha (linha pivô) de forma a transformar u_ℓ (elemento pivô) em 1 e todos os demais elementos da coluna pivô em 0.

Exemplo 1

A região viável tem 5 vértices, A , B , C , D e E são mostrados na figura abaixo:



Exemplo 1

Acrescentando as variáveis de folga x_4 , x_5 e x_6 , temos o seguinte problema na forma padrão:

$$\begin{array}{ll} \text{minimizar} & -10x_1 - 12x_2 - 12x_3 \\ \text{sujeita a} & x_1 + 2x_2 + 2x_3 + x_4 = 20, \\ & 2x_1 + x_2 + 2x_3 + x_5 = 20, \\ & 2x_1 + 2x_2 + x_3 + x_6 = 20, \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0. \end{array}$$

Exemplo 1

O vetor $x = (0, 0, 0, 20, 20, 20)$ é uma solução básica viável (ponto A da figura) e pode ser usado como ponto inicial para nosso algoritmo.

Vamos definir, então, $B(1) = 4$, $B(2) = 5$ e $B(3) = 6$. A matriz base correspondente é a matriz identidade.

Para calcular a linha 0 do tableau inicial, temos $c_B = 0$ e, portanto, $c_B^T x_B = 0$ e $\bar{c} = c$.

Exemplo 1

Portanto, o tableau inicial é

	x_1	x_2	x_3	x_4	x_5	x_6
	0	-10	-12	-12	0	0
$x_4 =$	20	1	2	2	1	0
$x_5 =$	20	2	1	2	0	1
$x_6 =$	20	2	2	1	0	1

Colocamos rótulos nas colunas para indicar que coluna está associada a qual variável. Fazemos o mesmo com as linhas, para indicar quais variáveis são básicas e qual a ordem usada.

Exemplo 1

O custo reduzido da variável x_1 é negativo e por isso esta variável será escolhida para entrar na base ($j = 1$).

A coluna pivô (associada à variável x_1) é $u = (1, 2, 2)$.

Calculamos então as razões $x_{B(i)}/u_i$, para $i = 1, 2, 3$. Para o nosso caso, temos

$$\frac{x_{B(1)}}{u_1} = \frac{x_4}{u_1} = \frac{20}{1} = 20, \quad \frac{x_{B(2)}}{u_2} = \frac{x_5}{u_2} = \frac{20}{2} = 10$$

$$\frac{x_{B(3)}}{u_3} = \frac{x_6}{u_3} = \frac{20}{2} = 10.$$

A menor razão (10) é dada pelos índices $i = 2$ e $i = 3$. Escolhemos $\ell = 2$.

Exemplo 1

Isso determina o elemento pivô (linha $\ell = 2$ e coluna $j = 1$), sublinhado no tableau abaixo.

	x_1	x_2	x_3	x_4	x_5	x_6
0	-10	-12	-12	0	0	0
$x_4 =$	20	1	2	1	0	0
$x_5 =$	20	<u>2</u>	1	0	1	0
$x_6 =$	20	2	2	1	0	1

Como $\ell = 2$, a segunda variável básica ($B(2) = 5$) deixa a base. A nova base é dada por $\bar{B}(1) = 4$, $\bar{B}(2) = j = 1$ e $\bar{B}(3) = 6$.

Exemplo 1

Para atualizar o tableau, precisamos somar um múltiplo da linha pivô ($\ell = 2$) em todas as outras linhas, de forma a obter custo reduzido 0 para a variável $x_j = x_1$ e a coluna $j = 1$ se tornar a ℓ -ésima coluna da identidade.

Para isso,

- multiplicamos a linha pivô por 5 e a somamos à linha 0, fazendo com que o custo reduzido de x_1 seja 0;
- multiplicamos a linha pivô por $1/2$ e a somamos na primeira linha, fazendo com que o primeiro elemento da primeira coluna se torne 0;
- subtraímos da linha pivô da terceira linha, fazendo com que o terceiro elemento da primeira coluna se torne 0;
- dividimos os elementos da linha pivô por 2, para que o segundo elemento da primeira coluna se torne 1.

Exemplo 1

Assim, o tableau atualizado fica

	x_1	x_2	x_3	x_4	x_5	x_6	
	100	0	-7	-2	0	5	0
$x_4 =$	10	0	1.5	1	1	-0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	-1	0	-1	1

A solução básica viável correspondente é $x = (10, 0, 0, 10, 0, 0)$ (ponto D da figura), com custo -100 . Note que, como x_6 é uma variável básica nula, x é degenerada.

Exemplo 1

Como mencionado anteriormente, o tableau indica que as restrições de igualdade podem ser escritas da maneira equivalente

$$\begin{cases} 10 = & 1.5x_2 + x_3 + x_4 - 0.5x_5 \\ 10 = x_1 + & 0.5x_2 + x_3 + 0.5x_5 \\ 0 = & x_2 - x_3 - x_5 + x_6 \end{cases}$$

Exemplo 1

Voltando ao tableau:

	x_1	x_2	x_3	x_4	x_5	x_6
100	0	-7	-2	0	5	0
$x_4 =$	10	0	1.5	1	-0.5	0
$x_1 =$	10	1	0.5	1	0.5	0
$x_6 =$	0	0	1	-1	-1	1

Temos que as variáveis x_2 e x_3 possuem custo reduzido negativo. Vamos escolher a variável x_3 ($j = 3$) para entrar na base.

Exemplo 1

A coluna pivô é a terceira: $u = (1, 1, -1)$. Como $u_3 < 0$, calculamos apenas $x_{B(1)}/u_1$ e $x_{B(2)}/u_2$.

Neste caso, temos

$$\frac{x_{B(1)}}{u_1} = \frac{x_4}{u_1} = \frac{10}{1} = 10, \quad \frac{x_{B(2)}}{u_2} = \frac{x_1}{u_2} = \frac{10}{1} = 10.$$

Novamente temos um empate e escolhemos $\ell = 1$, fazendo com que a variável básica $x_{B(\ell)} = x_{B(1)} = x_4$ saia da base.

Exemplo 1

O elemento pivô (linha $\ell = 1$ e coluna $j = 3$) está sublinhado no tableau:

	x_1	x_2	x_3	x_4	x_5	x_6	
	100	0	-7	-2	0	5	0
$x_4 =$	10	0	1.5	<u>1</u>	1	-0.5	0
$x_1 =$	10	1	0.5	1	0	0.5	0
$x_6 =$	0	0	1	-1	0	-1	1

Usamos agora a linha pivô (primeira linha) para atualizar o tableau.

Exemplo 1

Depois da atualização, temos

	x_1	x_2	x_3	x_4	x_5	x_6	
	120	0	-4	0	2	4	0
$x_3 =$	10	0	1.5	1	1	-0.5	0
$x_1 =$	0	1	-1	0	-1	1	0
$x_6 =$	10	0	2.5	0	1	-1.5	1

A solução básica associada é $x = (0, 0, 10, 0, 0, 10)$ (ponto B da figura), com custo -120.

Agora, a única variável com custo reduzido negativo é x_2 , então ela é escolhida para entrar na base ($j = 2$).

Exemplo 1

A segunda coluna é dada por $u = (1.5, -1, 2.5)$. Calculando as razões $x_{B(i)}/u_i$, para $i = 1, 3$ (já que $u_2 < 0$), temos que a menor é dada por $B(\ell) = B(3) = 6$.

Assim, a variável x_6 sai da base e o elemento pivô (linha $\ell = 3$ e coluna $j = 2$) é sublinhado no tableau:

	x_1	x_2	x_3	x_4	x_5	x_6	
	120	0	-4	0	2	4	0
$x_3 =$	10	0	1.5	1	1	-0.5	0
$x_1 =$	0	1	-1	0	-1	1	0
$x_6 =$	10	0	<u>2.5</u>	0	1	-1.5	1

Exemplo 1

Usando a terceira linha para atualizar o tableau, obtemos

	x_1	x_2	x_3	x_4	x_5	x_6
136	0	0	0	3.6	1.6	1.6
$x_3 =$	4	0	0	1	0.4	-0.6
$x_1 =$	4	1	0	-0.6	0.4	0.4
$x_2 =$	4	0	1	0	0.4	0.4

A solução básica viável associada é $x^* = (4, 4, 4, 0, 0, 0)$ (ponto E da figura), com custo -136.

Note que todos os custos reduzidos são não-negativos. Portanto, x^* é a solução ótima.

Exemplo 1

Neste exemplo, o método simplex fez três mudanças de base para chegar na solução e percorreu os pontos $A - D - B - E$ da figura.

Com diferentes regras de pivotamento, um caminho diferente poderia ser percorrido.

Uma pergunta é: o Método Simplex poderia percorrer o caminho $A - D - E$, que envolve somente duas arestas, em apenas 2 iterações?

A resposta é não! A primeira e a última bases diferem em 3 colunas, então são necessárias pelo menos 3 iterações do Método Simplex para passar de uma a outra.

Exemplo 2

O exemplo a seguir mostra que o Método Simplex pode, de fato, ciclar.

Considere o problema dado pelo tableau

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
3	$-3/4$	20	$-1/2$	6	0	0	0
$x_5 =$	0	$1/4$	-8	-1	9	1	0
$x_6 =$	0	$1/2$	-12	$-1/2$	3	0	0
$x_7 =$	1	0	0	1	0	0	1

Exemplo 2

Vamos usar a seguinte regra de pivotamento:

- (a) Seleccionamos a variável não-básica com custo reduzido mais negativo para entrar na base.
- (b) De todas as variáveis básicas que podem sair da base, seleccionamos a que tem menor índice.

Exemplo 2

O pivô está sublinhado e as atualizações do tableau são apresentadas a seguir:

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
3	$-3/4$	20	$-1/2$	6	0	0	0
$x_5 =$	0	<u>$1/4$</u>	-8	-1	9	1	0
$x_6 =$	0	$1/2$	-12	$-1/2$	3	0	1
$x_7 =$	1	0	0	1	0	0	1

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
3	0	-4	$-7/2$	33	3	0	0
$x_1 =$	0	1	-32	-4	36	4	0
$x_6 =$	0	0	4	$3/2$	-15	-2	1
$x_7 =$	1	0	0	1	0	0	1

Exemplo 2

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_1 =$	3	0	-4	$-7/2$	33	3	0	0
$x_6 =$	0	1	-32	-4	36	4	0	0
$x_7 =$	0	0	<u>4</u>	$3/2$	-15	-2	1	0
	1	0	0	1	0	0	0	1

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_1 =$	3	0	0	-2	18	1	1	0
$x_2 =$	0	1	0	8	-84	-12	8	0
$x_7 =$	0	0	1	$3/8$	$-15/4$	$-1/2$	$1/4$	0
	1	0	0	1	0	0	0	1

Exemplo 2

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_1 =$	3	0	0	-2	18	1	1	0
	0	1	0	<u>8</u>	-84	-12	8	0
$x_2 =$	0	0	1	$3/8$	$-15/4$	$-1/2$	$1/4$	0
$x_7 =$	1	0	0	1	0	0	0	1

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_3 =$	3	$1/4$	0	0	-3	-2	3	0
	0	$1/8$	0	1	$-21/2$	$-3/2$	1	0
$x_2 =$	0	$-3/64$	1	0	$3/16$	$1/16$	$-1/8$	0
$x_7 =$	1	$-1/8$	0	0	$21/2$	$3/2$	-1	1

Exemplo 2

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_3 =$	3	$1/4$	0	0	-3	-2	3	0
$x_2 =$	0	$1/8$	0	1	$-21/2$	$-3/2$	1	0
$x_7 =$	0	$-3/64$	1	0	$3/16$	$1/16$	$-1/8$	0
	1	$-1/8$	0	0	$21/2$	$3/2$	-1	1

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_3 =$	3	$-1/2$	16	0	0	-1	1	0
$x_4 =$	0	$-5/2$	56	1	0	2	-6	0
$x_7 =$	0	$-1/4$	$16/3$	0	1	$1/3$	$-2/3$	0
	1	$5/2$	-56	0	0	-2	6	1

Exemplo 2

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_3 =$	3	$-1/2$	16	0	0	-1	1	0
	0	$-5/2$	56	1	0	<u>2</u>	-6	0
$x_4 =$	0	$-1/4$	$16/3$	0	1	$1/3$	$-2/3$	0
$x_7 =$	1	$5/2$	-56	0	0	-2	6	1

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_5 =$	3	$-7/4$	44	$1/2$	0	0	-2	0
	0	$-5/4$	28	$1/2$	0	1	-3	0
$x_4 =$	0	$1/6$	-4	$-1/6$	1	0	$1/3$	0
$x_7 =$	1	0	0	1	0	0	0	1

Exemplo 2

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_5 =$	3	$-7/4$	44	$1/2$	0	0	-2	0
$x_4 =$	0	$-5/4$	28	$1/2$	0	1	-3	0
$x_7 =$	1	0	0	1	0	0	<u>$1/3$</u>	0
							0	1

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
$x_5 =$	3	$-3/4$	20	$-1/2$	6	0	0	0
$x_6 =$	0	$1/4$	-8	-1	9	1	0	0
$x_7 =$	0	$1/2$	-12	$-1/2$	3	0	1	0
	1	0	0	1	0	0	0	1

Exemplo 2

Fazendo as atualizações, chegamos exatamente ao mesmo tableau inicial.

Note que em todas as mudanças de base, tivemos $\theta^* = 0$.

Em particular, em todos tableaus intermediários tivemos a mesma solução básica viável $x = (0, 0, 0, 0, 0, 0, 1)$ e, conseqüentemente, mesmo custo ($c^T x = -3$).

Comparação entre Métodos Simplex revisado e tableau completo

Suponha que fossemos resolver o problema aumentado

$$\begin{array}{ll} \text{minimizar} & c^T x + 0^T y \\ \text{sujeita a} & Ax + Iy = b, \\ & x, y \geq 0. \end{array}$$

Para isso, poderíamos usar uma versão do Método Simplex que nunca permita que uma componente de y se torne básica. Então, o Método Simplex faz as mudanças de base como se o vetor y não existisse.

Comparação entre Métodos Simplex revisado e tableau completo

Note também que o vetor de custos reduzidos para este problema aumentado é

$$(c^T \mid 0^T) - c_B^T B^{-1}(A \mid I) = (\bar{c}^T \mid -c_B^T B^{-1}).$$

Então, o simplex tableau para o problema aumentado

$-c_B^T B^{-1}b$	\bar{c}^T	$-c_B^T B^{-1}$
$B^{-1}b$	$B^{-1}A$	B^{-1}

Comparação entre Métodos Simplex revisado e tableau completo

Em particular, seguindo o método do tableau completo, a inversa da matriz base B^{-1} é calculada em cada iteração.

Então, podemos pensar no Método Simplex revisado como sendo o mesmo que o tableau completo aplicado ao problema aumentado, com exceção de que a parte do tableau que armazena $B^{-1}A$ nunca é calculada explicitamente.

Em vez disso, apenas quando uma variável x_j é escolhida para entrar na base, a coluna pivô $B^{-1}A_j$ é computada.

Comparação entre Métodos Simplex revisado e tableau completo

Portanto, o Método Simplex revisado é apenas uma variante do tableau completo, mas com armazenamento mais eficiente.

Se o Método Simplex revisado também calcular as entradas da linha 0 que ficam acima de B^{-1} (usando operações elementares de linhas), os multiplicadores simplex $p^T = c_B^T B^{-1}$ são disponibilizados, eliminando a necessidade de resolver o sistema linear $p^T B = c_B^T$ a cada iteração.

Comparação entre Métodos Simplex revisado e tableau completo

Vejamos agora o número de operações que cada uma das implementações faz por iteração e suas necessidades de armazenamento computacional.

O tableau completo precisa de um número constante (e pequeno) de operações aritméticas para atualizar cada entrada do tableau.

Então, o número de operações computacionais por iteração é proporcional ao tamanho do tableau, que é $O(mn)$.

Comparação entre Métodos Simplex revisado e tableau completo

O Método Simplex revisado usa operações parecidas para atualizar B^{-1} e $c_B^T B^{-1}$ e, como somente $O(m^2)$ entradas são atualizadas, são necessárias $O(m^2)$ operações por iteração.

Além disso, o custo reduzido de cada variável x_j pode ser calculado usando o produto interno $p^T A_j$, que requer $O(m)$ operações.

No pior caso, é necessário calcular o custo reduzido de todas as variáveis, o que dá um custo total de $O(mn)$ operações por iteração.

Como $m \leq n$, o custo computacional no pior caso, por iteração, é $O(mn + m^2) = O(mn)$ para ambas as implementações.

Comparação entre Métodos Simplex revisado e tableau completo

No entanto, se considerarmos a regra de pivotamento que avalia um custo reduzido por vez, até que um custo reduzido negativo seja encontrado, uma iteração típica do Método Simplex revisado pode usar muito menos operações.

No melhor caso, se o primeiro custo reduzido é negativo e a primeira variável é escolhida para entrar na base, somente $O(m^2)$ são realizadas.

A conclusão é que o Método Simplex revisado não pode ser mais lento que o tableau completo. E pode ser muito mais rápido.

Comparação entre Métodos Simplex revisado e tableau completo

Outra vantagem importante do Método Simplex revisado em relação ao tableau completo é o gasto de memória.

O Método Simplex revisado precisa de $O(m^2)$ posições de memória, enquanto o tableau completo precisa de $O(mn)$. Quando n é muito maior do que m , esta diferença é muito grande.

Comparação entre Métodos Simplex revisado e tableau completo

Pode-se argumentar que o gasto de memória do Método Simplex revisado também é $O(mn)$, já que é necessário armazenar a matriz A . No entanto, em boa parte dos problemas de grande porte, a matriz A é esparsa e pode ser armazenada de maneira mais compacta.

Note que o fato de A ser esparsa não necessariamente ajuda no armazenamento necessário para o tableau completo, já que mesmo com A esparsa, $B^{-1}A$, em geral, não é.

Comparação entre Métodos Simplex revisado e tableau completo

	Tableau completo	Simplex revisado
Memória	$O(mn)$	$O(m^2)$
Tempo no pior caso	$O(mn)$	$O(mn)$
Tempo no melhor caso	$O(mn)$	$O(m^2)$