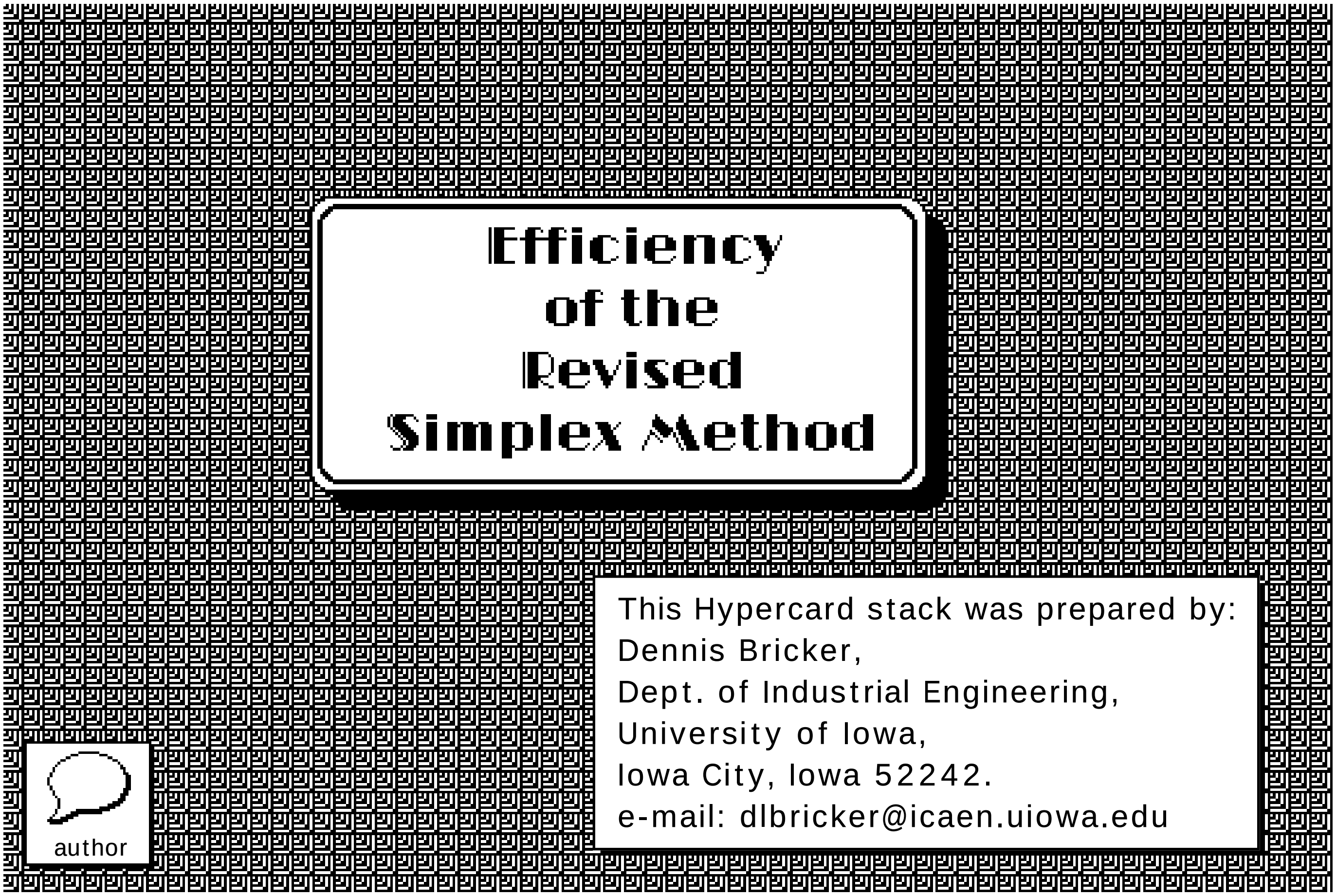
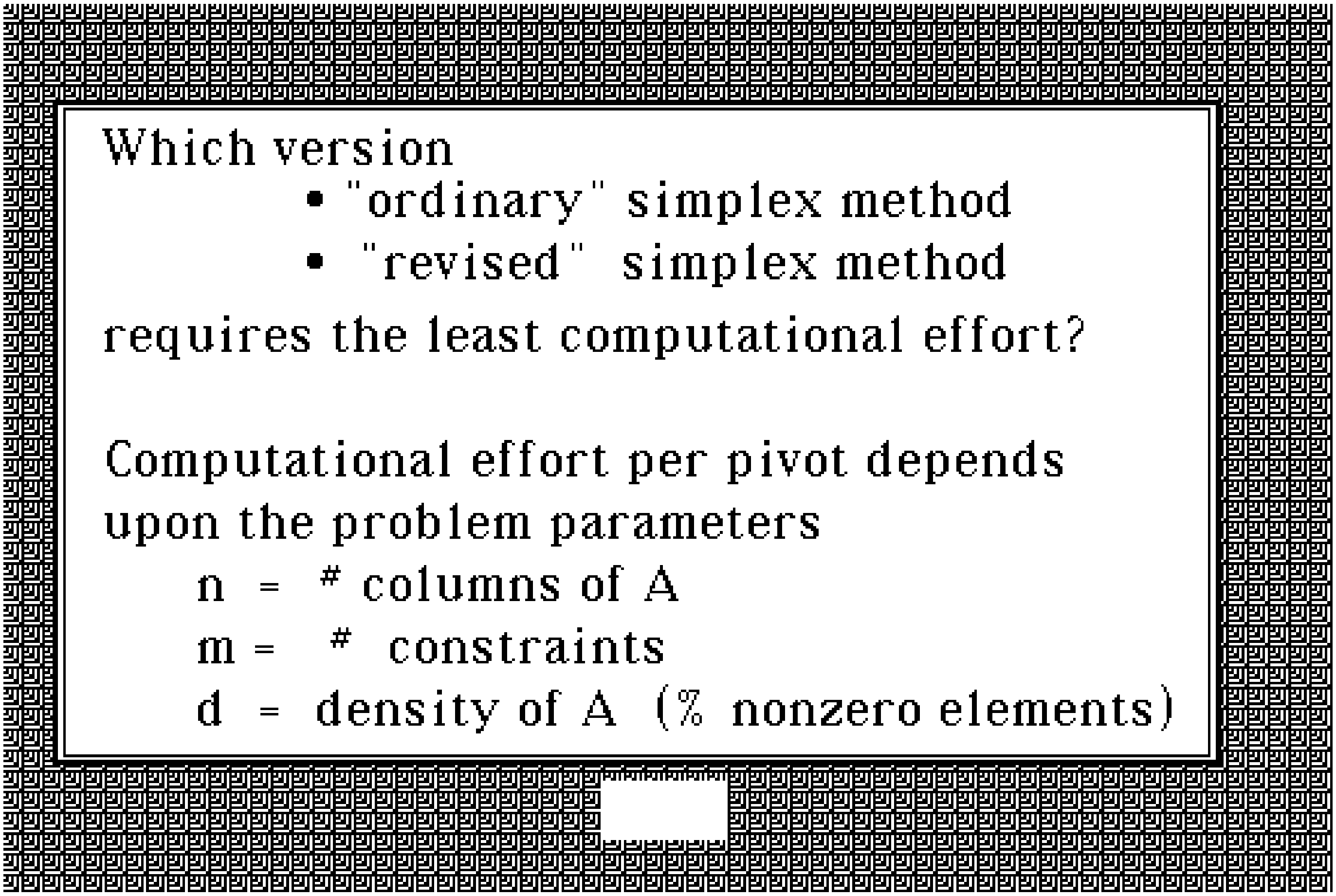


Efficiency of the Revised Simplex Method

This Hypercard stack was prepared by:
Dennis Bricker,
Dept. of Industrial Engineering,
University of Iowa,
Iowa City, Iowa 52242.
e-mail: dlbricker@icaen.uiowa.edu



Which version

- "ordinary" simplex method
- "revised" simplex method

requires the least computational effort?

Computational effort per pivot depends upon the problem parameters

n = # columns of A
m = # constraints
d = density of A (% nonzero elements)

Assume that, in the ordinary simplex tableau, previous pivots have increased the density such that we cannot make good use of sparse matrix techniques.

Let's count the number of
multiplications
& divisions
per pivot.

Consider the operations in a pivot in row r, column s:

$$\left[\begin{array}{cccccc|c}
\widehat{C}^1 & \widehat{C}^2 & \cdots & \widehat{C}^s & \cdots & \widehat{C}^n & \widehat{Z} \\
\hline
\widehat{A}_1^1 & \widehat{A}_1^2 & \cdots & \widehat{A}_1^s & \cdots & \widehat{A}_1^n & \widehat{b}_1 \\
\widehat{A}_2^1 & \widehat{A}_2^2 & \cdots & \widehat{A}_2^s & \cdots & \widehat{A}_2^n & \widehat{b}_2 \\
\vdots & \vdots & & \vdots & & \vdots & \vdots \\
\widehat{A}_r^1 & \widehat{A}_r^2 & \cdots & \enclose{circle}{\widehat{A}_r^s} & \cdots & \widehat{A}_r^n & \widehat{b}_r \\
\vdots & \vdots & & \vdots & & \vdots & \vdots \\
\widehat{A}_m^1 & \widehat{A}_m^2 & \cdots & \widehat{A}_m^s & \cdots & \widehat{A}_m^n & \widehat{b}_m
\end{array}\right]$$

➡ Ordinary Simplex Method

Pivoting in full tableau, with 100% density

➡ Revised Simplex Method

Explicit basis inverse maintained, and
density less than 100%

➡ Comparison of Algorithms

Ordinary Simplex Method

Operation Count
(x and ÷)
per iteration

❑ Minimum Ratio Test (pivot row selection)

m divisions

Pivot:

❑ Divide row r by $\widehat{A}_r^s$ *(need not divide in basic columns!)*

n−m divisions

⏎

□ For $i=1,2,\ldots m+1,\ i\neq r$,

add $-\widehat{A}_i^s$ times row r to row i

(only necessary to compute elements in nonbasic columns)

(n−m) multiplications per each of m rows

Ordinary Simplex Method

Total number of multiplications & divisions:

$$\begin{aligned} N_S &= m + (n-m) + m(n-m) \\ &= m + n + mn - m^2 \end{aligned}$$

per iteration.

⏎

Revised Simplex Method

Operation Count
(x and ÷)
per iteration

- ❑ Pricing each of (n-m) nonbasic columns $\tilde{C}^j = \pi A^j$
  (selecting pivot column)

  (dm) multiplications per each of (n-m) columns

- ❑ Computing substitution rates $\tilde{A}^s = (A^B)^{-1} A^j$
  (computing pivot column)

  dm multiplications per each of m rows

- ❑ Minimum ratio test (pivot row selection)

  m divisions

□ Pivot (update of basis inverse matrix, rhs, & π)

- divide row r of $\left(A^B\right)^{-1}$ & $\hat{b}$ by pivot element

  (m+1) divisions

- For i= 0, 1, 2, ...m (i = r):

  Add multiple of row r to row i

  (m+1) multiplications per each of m rows

Revised Simplex Method

Total number of multiplications & divisions:

$$N_R = dm(n-m) + dm^2 + m + (m+1) + (m+1)n$$

$$= dmn + m^2 + 3m + 1$$

per iteration.

Comparison of Algorithms

Multiplicatons & Divisions per iteration:

Ordinary Simplex $N_S = m + n + mn - m^2$

Revised Simplex $N_R = dmn + m^2 + 3m + 1$

Under what conditions is the revised simplex method more efficient that the ordinary simplex method?

That is, when is $N_R < N_S$?

⏎

$$N_R < N_S$$

$$\Longrightarrow dmn + m^2 + 3m + 1 < m + n + mn - m^2$$

$$\Longrightarrow dmn < mn + n - 2m^2 - 2m - 1$$

$$\Longrightarrow \quad d < 1 - 2\frac{m}{n} + \underbrace{\frac{1}{m} - \frac{2}{n} - \frac{1}{mn}}_{\textit{negligible}} \approx 1 - 2\frac{m}{n}$$

So the revised simplex method is more efficient than the ordinary simplex method when the density of the coefficient matrix A satisfies:

$$d < 1 - 2\frac{m}{n}$$

For example:

m	n	$1-2\frac{m}{n}$
10	50	60%
100	1000	80%
100	10000	98%

If m=10 & n=50, then the revised simplex method is more efficient if the density is less than about 60%

$$N_S = m + n + mn - m^2$$
$$N_R = dmn + m^2 + 3m + 1$$

For large LP problems in the "real world", the density is typically no more than 5%.

If m=100 and n=1000, $N_S = 91100$

	d=1%	d=5%
N_R	11301	15301
N_R/N_S	0.124	0.168

↩