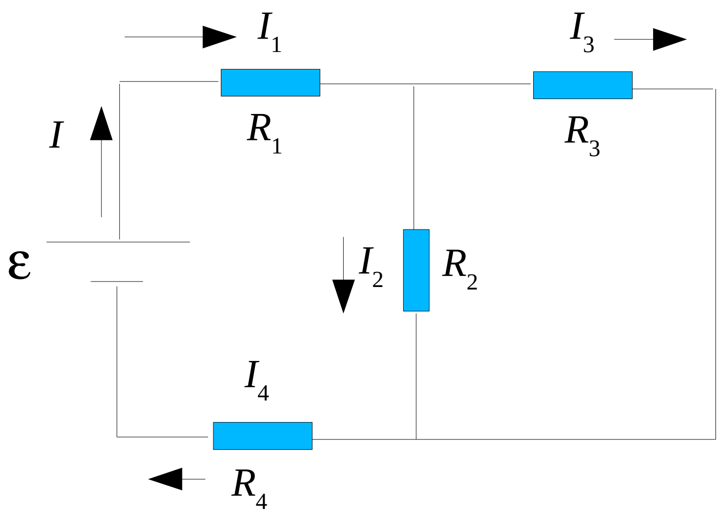
Solving Systems of Equations

A standard problem in physics is the multiple-loop circuit. For example,



i) use junction rule to eliminate extraneous subscripts on currents
   ($\Sigma\, I = 0$)

ii) pick a mini-circuit loop and apply $\Sigma\, V = 0$

iii) continue looping through mini-circuits until all elements
   have been covered

a) $\Sigma\, V = 0$:

b) $\Sigma\, V = 0$:

c) $\Sigma\, I = 0$:

   three equations, three unknowns

A general set of linear algebraic equations looks like

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1N}x_N = b_1$$
$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2N}x_N = b_2$$
$$\cdot$$
$$\cdot$$
$$\cdot$$
$$a_{M1}x_1 + a_{M2}x_2 + \ldots + a_{MN}x_N = b_M$$

where the x_i are unknown and everything else is known. If N=M, then a solution is likely. However, there may not be a unique solution if one or more of the M equations is a ______________________ of the others. This is called *row degeneracy*. A similar problem called *column degeneracy* can also crop up. A near degeneracy can also cause your program to fail if the round-off error is large enough.

In addition, if *N* is particularly large, the round-off errors may accumulate enough such that the program will also fail. Sophisticated algorithms have been developed to overcome these issues. A useful rule of thumb is that one can still use single precision and unsophisticated approaches if *N* is < 50 (if there is no degeneracy). *N* can approach several hundred and the program can still yield accurate results if the programmer resorts to double precision.

Matrix Review

In matrix form, the above set of linear equations can be expressed as

$$\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$$

$$\boldsymbol{A} = \begin{array}{l} a_{11}\ a_{12}\ \ldots\ a_{1N} \\ a_{21}\ a_{22}\ \ldots\ a_{2N} \\ . \\ . \\ . \\ a_{M1}\ a_{M2}\ \ldots\ a_{MN} \end{array} \qquad \mathbf{b} = \begin{array}{l} b_1 \\ b_2 \\ . \\ . \\ . \\ b_M \end{array} \qquad \mathbf{x} = \begin{array}{l} x_1 \\ x_2 \\ . \\ . \\ . \\ x_4 \end{array}$$

$\boldsymbol{x} = \boldsymbol{A}^{-1}\,\boldsymbol{b}$ where $\boldsymbol{A}^{-1}$ is the matrix inverse of $\boldsymbol{A}$