Iteration Methods for Linear Systems

- Many linear systems arised in real-world applications are large and sparse.
  - ⋄ Storage of data on a computer becomes a serious concern.
  - ⋄ Overhead in a direct method, such as the Gaussian elimination, usually becomes unbearable.
  - ⋄ Special techniques taking into account sparsity preservation, such as SPARSPAK, are available.
  - ⋄ Would like to solve $Ax = b$ with the matrix A intact and storage of only a few vectors.
- Motiviation of an Iterative Method:
  - ⋄ Let $\tilde{x}$ be an approximate solution to the system $Ax = b$.
  - ⋄ Define the *residual* by

    $$r := b - A\tilde{x}.$$

  - ⋄ The error $e := x - \tilde{x}$ satisfies the equation

    $$Ae = r. \tag{0.1}$$

  - ⋄ If we could solve (0.1) exactly, then $x := \tilde{x} + e$ would be the exact solution.
    - ▷ Note that solving (0.1) is as hard as the original problem.
    - ▷ Instead, we solve

      $$Se = r \tag{0.2}$$

      where S is an approximation to A.
    - ▷ The difference between A and S here is that (0.2) is much easier to be solved than (0.1).

  - ▷ Adding approximate correction to the approximate solution $\tilde{x}$ gives what we hope is a better approximate to the the true solution.

- A Prototype Iterative Algorithm:

  - ◇ x^{old}: = The current approximation to x;
  - ◇ Compute the residual $r := b - Ax^{\text{old}}$;
  - ◇ Solve $Se = r$ for the unknown e;
  - ◇ Set
$$x^{\text{new}} := x^{\text{old}} + e; \tag{0.3}$$
  - ◇ Repeat the cycle.

- Important Observation:

  - ◇ Multiplying (0.3) by S yields
$$\begin{aligned} Sx^{new} &= Sx^{old} + Se \\ &= Sx^{old} + b - Ax^{old} \\ &= (S - A)x^{old} + b := Tx^{old} + b \end{aligned} \tag{0.4}$$
where
$$A := S - T \tag{0.5}$$
is called a *splitting* of the matrix A.
  - ◇ If the iterates converges to a limit x, then $x = x^{old} = x^{new}$ and, by (0.4), we see that $Ax = b$. In other words, a limit point of the iteration scheme (0.4) is a solution of the original system.
  - ◇ It remains to determine the splitting so that
    - ▷ The sequence generated will converge.
    - ▷ The convergence rate is faster.