Chapter 11

Numerical Computation

- Continuous versus discrete
- Special features of the ODE's
- Challenges
- New directions.

Continuous versus Discrete

- The DE of a bridge can be integrated by any available ODE solvers.
  - ◇ ODE (Shampine and Gordon, '75).
  - ◇ MATLAB ODE SUITE (Shampine and Reichelt, '95).
  - ◇ RK (Sanz-Serna, '88).
- The discretization of a bridge could lead to a new iterative scheme.

- Sometimes the Poincaré map of the DE solution corresponds to the results of an existing iterative method.
  - ◇ Toda lattice and QR algorithm.
  - ◇ QZ flow and QZ algorithm.
  - ◇ SVD flow and SVD algorithm.
- Sometimes an existing discrete method may be extended directly into a continuous model.
  - ◇ Continuous Newton's method.
- Sometimes the underlying geometry suggests insights into the design of new iterative methods.
  - ◇ Newton's method for the inverse eigenvalue problem.
  - ◇ Newton's method for the inverse singular value problem.

Special Features of the ODE's

- For gradient flows:
  - ◇ Only the asymptotically stable equilibria are needed to answer the original problem.
  - ◇ An explicit Lyapunov function is often available.
- Orbits are required to stay on a certain manifold, such as $\mathcal{O}(n)$.
- Solutions usually preserve isospectrality.

Challenges

- Challenge to current ODE techniques:
  - ◇ The size of the differential system can easily be large.
  - ◇ Classical methods, such as multistep methods or Runge-Kutta methods fail to preserve isospectrality or orthonormality.
- Challenge to new ODE techniques:
  - ◇ Need an ODE solver that can effectively approximate the asymptotically stable equilibrium point.
  - ◇ Need an ODE solver that can trace trajectories on a manifold constraint (DAE).
  - ◇ Parallel computation.
  - ◇ VLSI implementation.

New Directions

- Orthogonal integrators (Sanz-Serna, '88, Dieci et al., '94, A. Iserles et al, '95).
  - ◇ Gauss-Legendre RK methods.
- Projected unitary schemes (Gear, '88).
  - ◇ Project approximations computed by an arbitrary scheme into $\mathcal{O}(n)$:

$$\left.\begin{array}{c} Z^T Z = I + O(h^p) \\ Z = QR \\ R_{ii} > 0 \end{array}\right\} \Rightarrow R = I + O(h^p).$$

- Systolic implementation (Moonen et al., '95).
  - ◇ A signal flow graph can be derived for the evaluation involved in the vector field.
  - ◇ Pipelining the signal flow graph ⇒ Parallel computation.
- Neural network adaptation.
  - ◇ Neural network ⇒ Analog computation ⇒ Dynamical systems.
  - ◇ Matrix DE's ⇒ Neural-network-like structures.