Chapter 2

QR-Type Flows

- Overview
- Basic form
- General framework
- Governing flows
- Basic theorems
- Abstract QR decompositions
- Abstract QR-type algorithms
- Relation to classical algorithms
- Nonclassical examples

Overview

- Construct a flow in the space of all $n \times n$ real matrices.
- Every element in the flow has the same eigenvalues.
- The flow converges to a limit point which has a specified structure.

Notations

$$
\begin{aligned}
Gl(n) &:= \{n \times n \text{ real nonsingular matrices}\} \\
gl(n) &:= \{n \times n \text{ real matrices}\} \\
X_0 &:= \text{A given matrix in} gl(n) \\
M(X_0) &:= \{g^{-1} X_0 g | g \in Gl(n)\} \\
[A, B] &:= AB - BA \text{ (Lie bracket)} \\
T &:= \text{Subspace of } gl(n) \\
P_T &:= \text{Projection mapping from } gl(n) \text{ to } T
\end{aligned}
$$

Basic Form

- Generic flow:

$$\frac{dX}{dt} = [X, k(X)].$$

- Basic relationship:

  ◇

$$\begin{aligned} \frac{dg(t)}{dt} &= g(t)k(t) \\ g(0) &= I \end{aligned}$$

  ◇

$$\begin{aligned} \frac{dX(t)}{dt} &= [X(t), k(t)] \\ X(0) &= X_0 \end{aligned}$$

  ◇

$$X(t) = g(t)^{-1}X_0 g(t).$$

General Framework

- Subspace splitting of $gl(n)$:

$$gl(n) = T_1 + T_2.$$

  - ◇ T_1 and T_2 are subspaces of $gl(n)$.
  - ◇ This is a subspace decomposition only, not necessarily a subalgebra decomposition of $gl(n)$.
  - ◇ Given T_1, one may choose $T_2 = gl(n) - T_1$. This is not necessarily a direct sum decomposition.

Examples

- Toda flow:
  - ◇ T_1 = Subspace of skew symmetric matrices.
  - ◇
    $$k(X) := (X^-) - (X^-)^T.$$
- General flow:
  - ◇ T_1 = Arbitrary linear subspace.
  - ◇
    $$k(X) := \text{Projection of } X \text{ onto subspace } T_1.$$
  - ◇ Time-1 mapping of the solution still enjoys a QR-type algorithm.
- Application to structured eigenvalue problems:
  - ◇ A QR-type algorithm preserving the Hamiltonian structure exists for the Hamiltonian eigenvalue problem.

Governing Flows

- The governing flow:

$$\begin{aligned} \frac{dX(t)}{dt} &:= [X(t), P_1(X(t))] \\ X(0) &:= X_0 \end{aligned}$$

  ◇ $P_1 :=$ Projection onto T_1.

- The associated flows:

$$\begin{aligned} \frac{dg_1(t)}{dt} &:= g_1(t)P_1(X(t)) \\ g_1(0) &:= I \end{aligned}$$

  and

$$\begin{aligned} \frac{dg_2(t)}{dt} &:= P_2(X(t))g_2(t) \\ g_2(0) &:= I \end{aligned}$$

  ◇ $P_2 :=$ Projection onto T_2.

Basic Theorems

All flows enjoys three basic properties:

- Similarity Property
- Decomposition Property
- Reverse Property

Similarity Property

$$X(t) = g_1(t)^{-1} X_0 g_1(t) = g_2(t) X_0 g_2(t)^{-1}.$$

- Define $Z(t) = g_1(t) X(t) g_1(t)^{-1}$.
- Check

$$\begin{aligned}
\frac{dZ}{dt} &= \frac{dg_1}{dt} X g_1^{-1} + g_1 \frac{dX}{dt} g_1^{-1} + g_1 X \frac{dg_1^{-1}}{dt} \\
&= (g_1 P_1(X)) X g_1^{-1} \\
&\quad + g_1 (X P_1(X) - P_1(X) X) g_1^{-1} \\
&\quad + g_1 X (-P_1(X) g_1^{-1}) \\
&= 0.
\end{aligned}$$

- Thus $Z(t) = Z(0) = X(0) = X_0$.

Decomposition Property

$$exp(tX_0) = g_1(t)g_2(t).$$

- Trivially $exp(X_0 t)$ satisfies the IVP

$$\frac{dY}{dt} = X_0 Y, Y(0) = I.$$

- Define $Z(t) = g_1(t)g_2(t)$.
- Then $Z(0) = I$ and

$$\begin{aligned}
\frac{dZ}{dt} &= \frac{dg_1}{dt} g_2 + g_1 \frac{dg_2}{dt} \\
&= (g_1 P_1(X))g_2 + g_1(P_2(X)g_2) \\
&= g_1 X g_2 \\
&= X_0 Z \quad \text{(by Similarity Property).}
\end{aligned}$$

- By the uniqueness theorem in the theory of ordinary differential equations, $Z(t) = exp(X_0 t)$.

Reverse Property

$$exp(tX(t)) = g_2(t)g_1(t).$$

- By Decomposition Property,

$$\begin{aligned} g_2(t)g_1(t) &= g_1(t)^{-1}exp(X_0 t)g_1(t) \\ &= exp(g_1(t)^{-1}X_0 g_1(t)t) \\ &= exp(X(t)t). \end{aligned}$$

Abstract QR-type Decomposition

- In Lie theory, corresponding to a Lie algebra decomposition of $gl(n)$, there is a Lie group decomposition of $Gl(n)$ in the neighborhood of I.
- We have shown, corresponding to a subspace decomposition $gl(n) = T_1 + T_2$, every matrix in the neighborhood of I can still be written as the product of two nonsingular matrices, i.e.,

$$exp(X_0 t) = g_1(t) g_2(t).$$

- The product $g_1(t)g_2(t)$ will be called the abstract $g_1 g_2$ decomposition of $exp(X_0 t)$.

Abstract QR-type Algorithm

- By setting $t = 1$ in Theorems 2 and 3, we have

$$\begin{aligned} exp(X(0)) &= g_1(1)g_2(1) \\ exp(X(1)) &= g_2(1)g_1(1). \end{aligned}$$

- Since the differential equation for $X(t)$ is autonomous, the above phenomenon will occur at every feasible integer time.

- Corresponding to the abstract g_1g_2 decomposition, the above iterative process for all feasible integers will be called the abstract g_1g_2 algorithm.

Relation to Classical Algorithms

	Case 1	Case 2	Case 3
T_1	$o(n)$	$l(n)$	$l(n)+d(n)/2$
T_2	$r(n)+d(n)$	$r(n)+d(n)$	$r(n)+d(n)/2$
$k(t)=P_1(X(t))$	$X^- - X^{-T}$	X^-	$X^- + X^0/2$
$P_2(X(t))$	$X^+ + X^0 + X^{-T}$	$X^+ + X^0$	$X^+ + X^0/2$
$g_1(t)$	$\mathrm{Q}(t)\in O(n)$	$\mathrm{L}(t)\in L(n)$	$\mathrm{G}(t)\in L(n)$
$g_2(t)$	$\mathrm{R}(t)\in R(n)$	$\mathrm{U}(t)\in R(n)$	$\mathrm{H}(t)\in R(n)$
Algorithm	QR	LU	Cholesky

$$
\begin{aligned}
o(n) &:= \{\text{Skew-symmetric matrices in } gl(n)\} \\
O(n) &:= \{\text{Orthogonal matrices in } Gl(n)\} \\
r(n) &:= \{\text{Strictly upper triangular matrices in } gl(n)\} \\
R(n) &:= \{\text{Upper triangular matrices in } Gl(n)\} \\
l(n) &:= \{\text{Strictly lower triangular matrices in } gl(n)\} \\
L(n) &:= \{\text{Lower triangular matrices in } Gl(n)\} \\
d(n) &:= \{\text{Diagonal matrices in } Gl(n)\}
\end{aligned}
$$

$$
\begin{aligned}
X^+ &:= \text{The strictly upper triangular matrix of } X \\
X^o &:= \text{The diagonal matrix of } X \\
X^- &:= \text{The strictly lower triangular matrix of } X
\end{aligned}
$$

Nonclassical Examples

- Assume:

$$
\begin{aligned}
X_0 &:= \text{symmetric} \\
\Delta &:= \text{Active index subset} \\
\hat{X}(t) &:= \text{Portion of } X(t) \text{ conforming to } \Delta \\
P_1(X(t)) &:= \hat{X}(t) - (\hat{X}(t))^T \\
P_2(X(t)) &:= X(t) - P_1(X(t))
\end{aligned}
$$

  Then:

$$\text{For all } (i,j) \in \Delta,\ x_{ij}(t) \longrightarrow 0 \text{ as } t \longrightarrow \infty.$$

  ◇ The above result suggests a way to produce (or knock out) any prescribed pattern that is symmetric to the diagonal of a symmetric matrix.

  ◇ The above result may be interpreted as a generalization of the Schur decomposition theorem (which knocks out the entire off-diagonal elements) for symmetric matrices.

  ◇ When $\Delta = \{(i,j) | 1 \leq j < i - 1 \leq n - 1\}$, the dynamical system represents a continuous tridiagonalization process.

- Assume

$$\begin{aligned} X_0 &:= \text{general (distinct eigenvalues)} \\ \Delta &\subset \{(i,j) | 1 \le j < i \le n\} \\ &:= \text{a } \mathit{rectangular} \text{ index subset} \\ \hat{X}(t) &:= \text{Portion of } X(t) \text{ conforming to } \Delta \\ P_1(X(t)) &:= \hat{X}(t) - (\hat{X}(t))^T \\ P_2(X(t)) &:= X(t) - P_1(X(t)) \end{aligned}$$

Then

For all $(i,j) \in \Delta,\ x_{ij}(t) \longrightarrow 0$ as $t \longrightarrow \infty$.

◇ The above result remains true if Δ is such that its complement represents a block upper triangular matrix. In this case, we have a continuous realization of the so called treppeniteration.

- Assume

$$\begin{aligned} X_0 &:= \text{Hamiltonian } \in gl(2n) \\ &:= \begin{bmatrix} A_0, & N_0 \\ K_0, & -A_0^T \end{bmatrix} \\ K, N &:= \text{symmetric } \in gl(n) \\ P_1(X(t)) &:= \begin{bmatrix} 0, & -K(t) \\ K(t), & 0 \end{bmatrix} \end{aligned}$$

Then

$a)$ $[X, P_1(X)]$ is Hamiltonian
$b)$ g_1 is both orthogonal and sympletic
$c)$ $X(t)$ remains Hamiltonian
$d)$ $K(t) \longrightarrow 0$ as $t \longrightarrow \infty$.

◇ The Hamiltonian eigenvalue problem for X_0 practically becomes the eigenvalue problem for

$$\lim_{t \to \infty} A(t).$$

◇ No explicit iterative scheme is known for the Hamiltonian eigenvalue problem due to the lack of knowledge of the structure of $g_2(t)$ in the abstract decomposition of $exp(X_0 t)$.