AM213B Numerical Methods for the Solution of Differential Equations

Lecture 05

List of topics in this lecture

- Consistency of LMM, characteristic polynomials, consistency condition
- Stability and zero-stability of LMM, root condition
- Stability theorem: relation between stability and zero-stability
- Dahlquist equivalence theorem: consistency + stability = convergence
- Stiff problems, very different time scales in a problem

Consistency of LMM (linear multi-step methods)

Local truncation error of an LMM is defined the same way as before:

Local truncation error

= residual term when substituting an exact solution into numerical method.

$$e_n(h) = \sum_{j=0}^{r} \alpha_j u(t_n + jh) - h \sum_{j=0}^{r} \beta_j f\left(u(t_n + jh), t_n + jh\right)$$

Consistency of LMM is defined the same way as before:

An LMM is consistent if $e_n(h) = O(h^{p+1})$ with $p > 0$.

Consistency condition

We investigate the condition on coefficients $\{\alpha_j, \beta_j\}$ for consistency.

We introduce characteristic polynomials of an LMM (there are two of them).

Recall the general form of LMM

$$\sum_{j=0}^{r} \alpha_j u_{n+j} = h \sum_{j=0}^{r} \beta_j f(u_{n+j}, t_{n+j}), \qquad \alpha_r = 1$$

Definition:

The two characteristic polynomials of an LMM are defined as

$$\rho(\xi)=\sum_{j=0}^{r}\alpha_j\xi^j \quad \leftarrow \text{ coefficients on the left side of LMM}$$

$$\sigma(\xi)=\sum_{j=0}^{r}\beta_j\xi^j \quad \leftarrow \text{ coefficients on the right side of LMM}$$

Theorem (consistency condition):

An LMM is consistent if and only if

$$\begin{cases}\rho(1)=0\\ \rho'(1)=\sigma(1)\end{cases}$$

Proof:

Recall that consistency means O(1) and O(h) terms disappear in the local truncation error. In the expression of local truncation error, we do Taylor expansion around t_n and calculate coefficients of terms up to O(h).

$$e_n(h)=\sum_{j=0}^{r}\alpha_j u(t_n+jh)-h\sum_{j=0}^{r}\beta_j f\left(u(t_n+jh),t_n+jh\right)$$

$$=\sum_{j=0}^{r}\alpha_j\left(u(t_n)+u'(t_n)jh\right)-h\sum_{j=1}^{r}\beta_j f\left(u(t_n),t_n\right)+O\left(h^2\right)$$

$$=u(t_n)\sum_{j=0}^{r}\alpha_j+h\left(u'(t_n)\sum_{j=0}^{r}\alpha_j j-f\left(u(t_n),t_n\right)\sum_{j=1}^{r}\beta_j\right)+O\left(h^2\right)$$

Use $f\left(u(t_n),t_n\right)=u'(t_n)$, $\sum_{j=0}^{r}\alpha_j=\rho(1)$, $\sum_{j=0}^{r}\alpha_j j=\rho'(1)$, $\sum_{j=1}^{r}\beta_j=\sigma(1)$

$$=u(t_n)\rho(1)+hu'(t_n)\left(\rho'(1)-\sigma(1)\right)+O(h^2)$$

Thus, $e_n(h)=O(h^2)$ if and only if the characteristic polynomials satisfy

$$\begin{cases}\rho(1)=0\\ \rho'(1)=\sigma(1)\end{cases}$$

Stability of LMM

Recall the stability we introduced earlier for single-step methods $u_{n+1}=L_{num}(u_n)$:

$$\left|L_{num}(u_n)-L_{num}(v_n)\right|\le(1+C\cdot h)\left|u_n-v_n\right| \quad \text{for small } h \qquad \text{(E01)}$$

where constant C is independent of h, u_n and v_n.

Applying the numerical method over n steps leads to

$$\left|\left(L_{num}\right)^n(u_0)-\left(L_{num}\right)^n(v_0)\right| \le C_T\left|u_0-v_0\right| \qquad \text{for small } h \text{ and } nh \le T \qquad \text{(E01B)}$$

where C_T is independent of h, n, u_0 and v_0 (as long as $nh \le T$).

(E01B) is more general in the sense that (E01) implies (E01B).

For an r-step LMM, we write it in the vector-operator form:

$$\vec{u}_{n+1} = L_{\text{LMM}}(\vec{u}_n)$$

$$\text{where} \qquad \vec{u}_n \equiv \begin{pmatrix} u_n \\ u_{n+1} \\ \vdots \\ u_{n+r-1} \end{pmatrix}, \qquad \vec{u}_0 \equiv \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{r-1} \end{pmatrix}$$

Applying the numerical method over n steps leads to

$$\vec{u}_n = \left(L_{\text{LMM}}\right)^n(\vec{u}_0)$$

While (E01) is convenient to use in analysis, it is <u>too narrow</u> and is <u>inadequate</u> for LMM. We demonstrate this in the simple example below.

<u>Example:</u>

We cast the <u>Euler method</u> into the form of an r–step LMM

$$u_{n+r} = u_{n+r-1} + \sum_{j=0}^{r} \beta_j f(u_{n+j}, t_{n+j}), \qquad \beta_j = \begin{cases} 1, & j = r-1 \\ 0, & \text{otherwise} \end{cases}$$

We apply it to ODE $u' = 0$. It becomes $u_{n+r} = u_{n+r-1}$ for $n \ge 0$.

In vector-operator form, this simple numerical method (for ODE $u' = 0$) is

$$\vec{u}_{n+1} = L_{\text{LMM}}(\vec{u}_n) = \begin{pmatrix} \vec{u}_n(2) \\ \vec{u}_n(3) \\ \vdots \\ \vec{u}_n(r) \\ \vec{u}_n(r) \end{pmatrix}$$

where $\vec{u}_n(i)$ denotes the i-th component of vector $\vec{u}_n$.

To examine the validity of (E01) and (E01B), we apply L_{LMM} to two special vectors:

$$\vec{u}_0 = (\ 0\ \ \cdots\ \ 0\ \ 1\ )^T, \qquad \vec{v}_0 = (\ 0\ \ \cdots\ \ 0\ )^T.$$

We have

$$L_{\text{LMM}}(\vec{u}_0) = (\ 0\ \ \cdots\ \ 0\ \ 1\ \ 1\ )^T$$

$$\left(L_{\text{LMM}}\right)^n(\vec{u}_0) = (\ 1\ \ \cdots\ \ 1\ \ 1\ )^T \qquad \text{for } n \geq (r-1)$$

$$\left(L_{\text{LMM}}\right)^n(\vec{v}_0) = 0$$

It follows that

- $\underbrace{\left\|\left(L_{\text{LMM}}\right)^n(\vec{u}_0) - \left(L_{\text{LMM}}\right)^n(\vec{v}_0)\right\|}_{\left\|(\ 1\ \cdots\ 1\ \ 1\ )^T - 0\right\|} \leq C_T \underbrace{\left\|\vec{u}_0 - \vec{v}_0\right\|}_{\left\|(\ 0\ \cdots\ 0\ \ 1\ )^T - 0\right\|}$ is true.
- $\underbrace{\left\|L_{\text{LMM}}(\vec{u}_0) - L_{\text{LMM}}(\vec{v}_0)\right\|}_{\left\|(\ 0\ \cdots\ 0\ \ 1\ \ 1\ )^T - 0\right\|} \leq (1 + C \cdot h) \underbrace{\left\|\vec{u}_0 - \vec{v}_0\right\|}_{\left\|(\ 0\ \cdots\ 0\ \ 1\ )^T - 0\right\|}$ is NOT true.

It is clear that this simple method (for ODE $u' = 0$) should be classified as "stable".

Therefore, <u>we select (E01B) to define the stability for LMM</u>.

<u>Definition of stability for LMM</u>

An LMM $\vec{u}_{n+1} = L_{\text{LMM}}(\vec{u}_n)$ is stable if

$$\left\|\left(L_{\text{LMM}}\right)^n(\vec{u}_0) - \left(L_{\text{LMM}}\right)^n(\vec{v}_0)\right\| \leq C_T \left\|\vec{u}_0 - \vec{v}_0\right\| \qquad \text{for small } h \text{ and } nh \leq T$$

where C_T is independent of h, n, u_0 and v_0 (as long as $nh \leq T$).

<u>Remark:</u>

This stability is <u>difficult to check</u> directly. We need to work with alternative conditions that are more convenient to check.

<u>Zero-stability of LMM</u>

Consider an LMM applied to the model ODE: $u' = 0$

<u>Exact solution:</u>

$u(t) = \text{const}$

<u>Numerical solution:</u>

$$\sum_{j=0}^{r} \alpha_j u_{n+j} = 0$$

Motivation:

We want numerical solution u_n to remain bounded as $n \to \infty$.

Definition of zero-stability:

If all solutions of $\sum_{j=0}^{r} \alpha_j u_{n+j} = 0$ remain bounded as $n \to \infty$,

then we say the corresponding LMM is zero-stable.

Remark:

The zero-stability is not affected by coefficients $\{\beta_j\}$.

Next we connect the zero-stability of an LMM to its characteristic polynomial $\rho(\xi)$.

In equation $\sum_{j=0}^{r} \alpha_j u_{n+j} = 0$, we consider solutions of the form $\{u_k = \xi^k, k=0, 1, 2, \ldots\}$.

Substituting $u_k = \xi^k$ into the equation, we get

$$\sum_{j=0}^{r} \alpha_j \xi^{n+j} = 0 \quad \longrightarrow \quad \sum_{j=0}^{r} \alpha_j \xi^{j} = 0$$

==> $\boxed{\rho(\xi) = 0}$

For polynomial $\rho(\xi)$, let

$\{ \xi_j \}$ denote simple roots and

$\{ q_i \}$ denote roots with multiplicity > 1.

The general numerical solution has the form:

$$u_n = \left(c_1 \xi_1{}^n + c_2 \xi_2{}^n + \cdots\right) + \left(b_1^{(0)} + b_1^{(1)} n + \cdots\right) q_1{}^n + \left(b_2^{(0)} + b_2^{(1)} n + \cdots\right) q_2{}^n + \cdots$$

u_n remains bounded as $n \to \infty$ if and only if

$| \xi_j | \le 1$ and $| q_i | < 1$

This is called the root condition.

Definition (root condition)

For a polynomial, if

- all roots satisfy $| \xi_j | \le 1$ and
- all roots with multiplicity above 1 satisfy $| q_i | < 1$,

then we say the polynomial satisfies the root condition.

We summarize the result we obtain above into a theorem.

Theorem (condition for zero-stability):

An LMM is zero-stable if and only if its characteristic polynomial $\rho(\xi)$ satisfies the root condition.

Next we look at the connection between the zero-stability and the stability.

Theorem (the stability theorem):

Suppose $f(u, t)$ in $u' = f(u, t)$ is Lipschitz continuous.

If an LMM is zero-stable, then it is stable.

That is, for any $T > 0$, there exists C_T such that

$$\left\| \left(L_{\text{LMM}}\right)^n (\vec{u}_0) - \left(L_{\text{LMM}}\right)^n (\vec{v}_0) \right\| \le C_T \left\| \vec{u}_0 - \vec{v}_0 \right\| \qquad \text{for small } h \text{ and } nh \le T$$

where C_T is independent of h, n, u_0 and v_0 (as long as $nh \le T$).

Proof: We will not go through the proof in details. An outline of key steps in the proof is presented in Appendix B.

This theorem connects the zero-stability to the stability. With this theorem as the stepping-stone, we now introduce the Dahlquist equivalence theorem (which is similar to the equivalence theorem we proved for single-step methods:

consistency + stability = convergence

Theorem (Dahlquist equivalence theorem)

An LMM method is convergent if and only if it is zero-stable and is consistent.

Proof: We will not go through the proof in details. The key steps for proving the Dahlquist equivalence theorem are similar to the steps used for proving the stability theorem (Appendix B). The adaptation of these steps for proving the Dahlquist equivalence theorem is discussed in Appendix C.

Example:

The midpoint method:

$$u_{n+2} - u_n = 2h f(u_{n+1}, t_{n+1})$$

It is a 2-step LMM. The two characteristic polynomials are

$$\rho(\xi) = \xi^2 - 1, \qquad \sigma(\xi) = 2\xi$$

Checking the consistency condition:

$\rho(1) = 0, \qquad \rho'(1) = 2 = \sigma(1)$

==> It is consistent.

Checking the root condition:

$\rho(\xi)$ has two simple roots:

$\xi_1 = 1, \qquad \xi_2 = -1$

==> It satisfies the root condition.

==> It is zero-stable.

By the Dahlquist equivalence theorem, the midpoint method is convergent.

Remark: In numerical simulations, we find that the midpoint method has an exponentially growing error mode even for the simple ODE: $u' = -u$. The exponentially growing error mode quickly ruins the numerical solution as time increases. This behavior does not contradict the Dahlquist equivalence theorem. At a finite time, if we use a very small time step, the midpoint method will be well behaved.

$$\begin{pmatrix} h \to 0 \\ \text{while } T \text{ is fixed} \end{pmatrix} \quad \text{vs} \quad \begin{pmatrix} T \to \infty \\ \text{while } h \text{ is fixed although small} \end{pmatrix}$$

Example:

All Adams methods (both Adams-Bashforth and Adams-Moulton) are zero-stable.

The general form of r-step Adams methods:

$$u_{n+r} = u_{n+r-1} + h\sum_{j=0}^{r} \beta_j f(u_{n+j}, t_{n+j})$$

Characteristic polynomial $\rho(\xi)$:

$$\rho(\xi) = \xi^r - \xi^{(r-1)} = \xi^{(r-1)}(\xi - 1)$$

$\rho(\xi)$ has one sample root and one root of multiplicity $(r - 1)$.

$\xi_1 = 1,$ a simple root,

$q_1 = 0,$ a root of multiplicity $(r - 1)$.

==> It satisfies the root condition.

==> It is zero-stable.

Question: Is zero-stability enough for making a method well behaved?

Consider numerical solutions of ODE

$$u' = -\lambda \sinh\big(u - \cos(t)\big), \qquad \lambda = \text{large} > 0 \qquad \text{(E02)}$$

In Assignment #1, you observed the results below.

> The numerical solution of Euler method is not even bounded unless the time step is tiny. The implicit trapezoidal method performs much better than Euler method. The implicit backward Euler is even better.

The bottom line:

> Zero-stability implies that when the time step is small enough, the numerical solution is well behaved. But for a certain category of problems, we cannot afford very tiny time steps. We need other types of stability to ensure that the numerical solution is well behaved even when the time step is not very small.

Stiff problems

We consider a linear version of (E02) as a model problem.

$$\begin{cases} u' = -\lambda\big(u - \cos(t)\big), & \lambda = \text{large} > 0 \\ u(0) = 0 \end{cases} \qquad \text{(E03A)}$$

Exact solution:

$$u(t) = \frac{\lambda^2}{1+\lambda^2}\cos(t) + \frac{\lambda}{1+\lambda^2}\sin(t) - \frac{\lambda^2}{1+\lambda^2}e^{-\lambda t}$$

(See Appendix A for derivation)

There are two very different time scales in this problem:

- Slow evolution of $\cos(t)$, and
- Very fast decay of $\exp(-\lambda t)$

Definition (stiff problem):

> A problem is called stiff if it has (at least) two very different time scales.

In ODE (E03A), the two time scales are tangled together. To simplify the discussion, we consider an even simpler problem in which the two time scales are separated.

$$\begin{cases} u' = -\lambda u, & \lambda = \text{large} > 0 \\ v' = -v \end{cases} \qquad \text{(E03B)}$$

We focus on the u-component of (E03B) and use it as a model problem for examining the performance of various numerical methods.

A simplified model:

$$\begin{cases} u' = -\lambda u, & \lambda = \text{large} > 0 \\ u(0) = 1 \end{cases}$$

Exact solution:

$$u(t) = \exp(-\lambda t)$$

Two properties of the exact solution:

1. $u(t)$ decreases to zero as $t \to \infty$.

   For large λ, $u(t)$ decreases very fast.

2. For a fixed value of $t > 0$ (no matter how small it is), $u(t) \to 0$ as $\lambda \to \infty$.

These are the properties we want to preserve in numerical solutions.

Behaviors of numerical solutions of $u' = -\lambda u$

Euler method:

$$u_{n+1} = u_n + h(-\lambda u_n)$$

==> $u_{n+1} = u_n(1 - \lambda h)$

==> $u_n = u_0(1 - \lambda h)^n$

As $n \to \infty$, u_n decreases in absolute value if and only if $|1 - \lambda h| < 1$,

if and only if $0 < \lambda h < 2$,

if and only if $h < \frac{2}{\lambda}$.

For large λ, this condition is very restrictive.

$\lambda = 10^8$ ==> $h < 2 \times 10^{-8}$

For u_n to decrease without oscillating in sign, we need $h \le \frac{1}{\lambda}$.

Conclusion for Euler method:

- For large λ, Euler method has to use a tiny time step just to ensure that the numerical solution decreases in absolute value as $n \to \infty$. In other words, the numerical solution is well behaved ONLY when the fast evolution component is resolved, which requires a tiny time step.
- In a stiff problem, the requirement of a tiny time step is coupled with the need of simulating the slow evolution component over a long time period.

  ==> Extremely large number of time steps is needed.

For example, to accommodate $\lambda = 10^8$ and to simulate to $T = 10$, we need

$h < 2/\lambda = 2\times10^{-8}$

$N = T/h > 5\times10^8$ **= 500 million time steps**

Backward Euler method:

$$u_{n+1} = u_n + h(-\lambda u_{n+1}), \qquad \lambda = \text{large} > 0$$

==> $u_{n+1}(1+\lambda h) = u_n$

==> $u_{n+1} = u_n \dfrac{1}{(1+\lambda h)}$

==> $u_n = u_0 \dfrac{1}{(1+\lambda h)^n}$

Property 1:

As $n \to \infty$, u_n decreases to 0 without oscillating in sign, for any value of $h > 0$.

Property 2:

When h is fixed, we have $u_1 = u_0 \dfrac{1}{(1+\lambda h)} \to 0$ as $\lambda \to \infty$.

Conclusion for Backward Euler method:

- Even for very large λ, the numerical solution of backward Euler method decreases to 0 without oscillating in sign, as $n \to \infty$, for any time step $h > 0$. The backward Euler method preserves both properties 1 and 2.
- In other words, the numerical solution is always well behaved even when the fast evolution component is not resolved. We can select the time step based on the need of resolving the slow evolution component.

Trapezoidal method:

$$u_{n+1} = u_n + \frac{h}{2}(-\lambda u_n - \lambda u_{n+1})$$

==> $u_{n+1}\left(1+\dfrac{\lambda h}{2}\right) = u_n\left(1-\dfrac{\lambda h}{2}\right)$

==> $u_{n+1} = u_n\left(\dfrac{1-\frac{\lambda h}{2}}{1+\frac{\lambda h}{2}}\right)$

$$\Longrightarrow \qquad u_n = u_0 \left(\frac{1-\frac{\lambda h}{2}}{1+\frac{\lambda h}{2}} \right)^n$$

The multiplication factor satisfies

$$\left| \frac{1-\frac{\lambda h}{2}}{1+\frac{\lambda h}{2}} \right| < 1 \qquad \text{for all values of } h > 0$$

Property 1:

As $n \to \infty$, u_n decreases in absolute value to 0, for any value of $h > 0$.

However, for $\lambda h > 2$, the multiplication factor is negative.

==> For $\lambda h > 2$, u_n oscillates in sign while decreasing to 0 as $n \to \infty$.

Property 2:

When h is fixed, we have $u_1 = \left(\frac{1-\frac{\lambda h}{2}}{1+\frac{\lambda h}{2}} \right) u_0 \to (-u_0)$ as $\lambda \to \infty$.

Conclusion for trapezoidal method:

- The numerical solution of trapezoidal method decreases to 0 as $n \to \infty$, for any time step $h > 0$. So the numerical solution will always remain bounded for any time step. We don't need to restrict the selection of time step to make the numerical solution bounded.
- However, for large λ, as $n \to \infty$, the numerical solution of trapezoidal method oscillates in sign with a very slow decay in amplitude. The trapezoidal method preserves property 1, but not property 2.

Based on the performances of Euler, backward Euler and trapezoidal methods analyzed above, we introduce the A-stability and the L-stability to measure the performance of a numerical method for solving stiff problems.

Intuitively, we define A-stability and L-stability as follows:

A-stability describes whether or not property 1 is preserved.

L-stability described whether or not both properties 1 & 2 are preserved.

Appendix A: Exact solution of the IVP

$$\begin{cases} u' = -\lambda\big(u - \cos(t)\big) \\ u(0) = 0 \end{cases}$$

We write the ODE as

$$u' + \lambda u = \lambda \cos(t)$$

We use the integrating factor method. Multiplying by $e^{\lambda t}$, leads to

$$e^{\lambda t} u' + \lambda e^{\lambda t} u = \lambda e^{\lambda t} \cos(t)$$

$$\Longrightarrow \quad \left(e^{\lambda t} u\right)' = \lambda e^{\lambda t} \cos(t)$$

Integrating from 0 to t, we get

$$e^{\lambda t} u(t) = \lambda \int_0^t e^{\lambda s} \cos(s)\, ds$$

We use the integration formula

$$\int_0^t e^{\lambda s} \cos(s)\, ds = \text{Real}\left[\int_0^t e^{(\lambda + i)s}\, ds\right] = \text{Real}\left[\frac{1}{\lambda + i}\left(e^{(\lambda + i)t} - 1\right)\right]$$

$$= \text{Real}\left[\frac{\lambda - i}{\lambda^2 + 1}\left(e^{\lambda t}\cos(t) - 1 + i e^{\lambda t}\sin(t)\right)\right] = \frac{\lambda}{\lambda^2 + 1}\left(e^{\lambda t}\cos(t) - 1\right) + \frac{1}{\lambda^2 + 1} e^{\lambda t}\sin(t)$$

The solution of the IVP is

$$u(t) = e^{-\lambda t}\lambda \int_0^t e^{\lambda s}\cos(s)\, ds$$

$$= \frac{\lambda^2}{1 + \lambda^2}\cos(t) + \frac{\lambda}{1 + \lambda^2}\sin(t) - \frac{\lambda^2}{1 + \lambda^2} e^{-\lambda t}$$

Appendix B: Proof of the stability theorem

Theorem (the stability theorem):

Suppose $f(u, t)$ in $u' = f(u, t)$ is Lipschitz continuous.

If an LMM is zero-stable, then for any $T > 0$, there exists C_T such that

$$\left\| \left(L_{\text{LMM}}\right)^n (\vec{u}_0) - \left(L_{\text{LMM}}\right)^n (\vec{v}_0) \right\| \le C_T \left\| \vec{u}_0 - \vec{v}_0 \right\| \qquad \text{for small } h \text{ and } nh \le T$$

where C_T is independent of h, n, u_0 and v_0 (as long as $nh \le T$).

Outline of key steps in the proof:

1. We first look at the LMM applied to solving $u' = 0$. It has the form

$$\sum_{j=0}^{r} \alpha_j u_{n+j} = 0 , \qquad \alpha_r = 1$$

In the matrix-vector form, we can write it as

$$\vec{u}_{n+1} = A\vec{u}_n$$

where vector u and matrix A are

$$\vec{u}_n \equiv \begin{pmatrix} u_n \\ u_{n+1} \\ \vdots \\ u_{n+r-1} \end{pmatrix}, \qquad A \equiv \begin{pmatrix} 0 & 1 & & 0 \\ \vdots & \ddots & \ddots & \\ 0 & \cdots & 0 & 1 \\ -\alpha_0 & -\alpha_1 & \cdots & -\alpha_{r-1} \end{pmatrix}$$

Matrix A is a constant matrix, independent of h. The characteristic polynomial of matrix A is $\rho(\xi)$. Polynomial $\rho(\xi)$ determines the zero-stability of the LMM. When the LMM is zero-stable, the eigenvalues of matrix A satisfy the root condition. By writing matrix A in the Jordan canonical form, we can show that there exists $C_A \ge 1$ such that

$$\left\| A^n \right\| \le C_A \qquad \text{for all } \; n \ge 0$$

2. Next, we look at the LMM applied to solving $u' = f(u, t)$. It has the form

$$\sum_{j=0}^{r} \alpha_j u_{n+j} = h \sum_{j=0}^{r} \beta_j f\left(u_{n+j}, t_{n+j}\right), \qquad \alpha_r = 1$$

For simplicity, we focus on explicit methods. We write it in the matrix-vector form

$$\vec{u}_{n+1} = A\vec{u}_n + h\vec{\phi}\left(\vec{u}_n, t_n\right) \qquad \text{(E05)}$$

where function $\phi(u, t)$ satisfies Lipschitz continuity with respect to u:

$$\left\| \vec{\phi}\left(\vec{u}, t\right) - \vec{\phi}\left(\vec{v}, t\right) \right\| \le C_L \left\| \vec{u} - \vec{v} \right\|$$

Note that while mapping Au is linear, function $\phi(u, t)$ is non-linear.

Let

$$\Delta\vec{u}_n \equiv \vec{u}_n - \vec{v}_n \quad \text{and} \quad \Delta\vec{\phi}_n \equiv \vec{\phi}\left(\vec{u}_n, t_n\right) - \vec{\phi}\left(\vec{v}_n, t_n\right)$$

The Lipschitz continuity gives us

$$\left\|\Delta\vec{\phi}_n\right\| \le C_L \left\|\Delta\vec{u}_n\right\|$$

Taking the difference of (E05) between *u* and *v* yields

$$\Delta\vec{u}_{n+1} = A\Delta\vec{u}_n + h\Delta\vec{\phi}_n \qquad \text{(E06)}$$

Substituting (E06) at *n*=0 into (E06) at *n* = 1, and then into (E06) at *n* = 2, … we obtain

$$\Delta\vec{u}_2 = A^2\Delta\vec{u}_0 + h\left(A\Delta\vec{\phi}_0 + \Delta\vec{\phi}_1\right)$$

$$\Delta\vec{u}_{n+1} = A^{n+1}\Delta\vec{u}_0 + h\sum_{j=0}^{n} A^{n-j}\Delta\vec{\phi}_j \qquad \text{(E07)}$$

Taking norm of both sides, using $\left\|A^n\right\| \le C_A$ and the Lipschitz continuity $\left\|\Delta\vec{\phi}_n\right\| \le C_L\left\|\Delta\vec{u}_n\right\|$, we arrive at a recursive inequality for $\left\|\Delta\vec{u}_n\right\|$

$$\left\|\Delta\vec{u}_{n+1}\right\| \le C_A\left\|\Delta\vec{u}_0\right\| + hC_AC_L\sum_{j=0}^{n}\left\|\Delta\vec{u}_j\right\| \qquad \text{(E08)}$$

3. Now we solve the recursive inequality (E08).

   We introduce {g_n} recursively as

$$g_0 = C_A\left\|\Delta\vec{u}_0\right\|$$

$$g_{n+1} = g_0 + hC_AC_L\sum_{j=0}^{n} g_j \qquad \text{(E09)}$$

   It is straightforward to show that

$$\left\|\Delta\vec{u}_n\right\| \le g_n \qquad \text{for all } n \ge 0$$

   To calculate g_n, we re-write the recursive equation (E09) as

$$g_{n+1} = g_n + hC_AC_Lg_n = (1+hC_AC_L)g_n$$

   Applying this relation successively from index 0 to index (*n*–1), yields

$$g_n = (1+hC_AC_L)^n g_0 \le \exp\left(C_AC_LT\right)\left\|\Delta\vec{u}_0\right\| \qquad \text{for } nh \le T$$

   Using $\left\|\Delta\vec{u}_n\right\| \le g_n$ and $g_0 = C_A\left\|\Delta\vec{u}_0\right\|$, we have

$$\left\|\Delta\vec{u}_n\right\| \le C_A\exp\left(C_AC_LT\right)\left\|\Delta\vec{u}_0\right\| \qquad \text{for } nh \le T$$

   Therefore, we finally arrive at

$$\left\|\left(L_{LMM}\right)^n\vec{u}_0 - \left(L_{LMM}\right)^n\vec{v}_0\right\| \le C_A\exp\left(C_AC_LT\right)\left\|\vec{u}_0 - \vec{v}_0\right\| \qquad \text{for } nh \le T$$

which implies the LMM is stable, by definition.

Appendix C: Key steps for proving Dahlquist equivalence theorem

Below we discuss briefly the key steps for proving the Dahlquist equivalence theorem. These steps are adapted from the steps in Appendix B for proving the stability theorem.

Again, we write the LMM in the matrix-vector form

$$\vec{u}_{n+1} = A\vec{u}_n + h\vec{\phi}\left(\vec{u}_n, t_n\right)$$

We introduce the vector versions of exact solution and local truncation error.

$$\vec{v}_n \equiv \vec{u}\left(t_n\right) = \begin{pmatrix} u\left(t_n\right) \\ u\left(t_{n+1}\right) \\ \vdots \\ u\left(t_{n+r-1}\right) \end{pmatrix}, \qquad \vec{e}_n\left(h\right) \equiv \begin{pmatrix} 0 \\ 0 \\ \vdots \\ e_n\left(h\right) \end{pmatrix} = O\left(h^{p+1}\right)$$

Numerical solution *u* and exact solution *v* satisfy, respectively

$$\begin{aligned} \vec{u}_{n+1} &= A\vec{u}_n + h\vec{\phi}\left(\vec{u}_n, t_n\right) \\ \vec{v}_{n+1} &= A\vec{v}_n + h\vec{\phi}\left(\vec{v}_n, t_n\right) + \vec{e}_n\left(h\right) \end{aligned} \qquad \text{(E25)}$$

Let $\Delta\vec{u}_n \equiv \vec{u}_n - \vec{v}_n$ and $\Delta\vec{\phi}_n \equiv \vec{\phi}\left(\vec{u}_n, t_n\right) - \vec{\phi}\left(\vec{v}_n, t_n\right)$. Taking the difference in (E25) gives us

$$\Delta\vec{u}_{n+1} = A\Delta\vec{u}_n + h\Delta\vec{\phi}_n - \vec{e}_n(h) \qquad \text{(E26)}$$

Substituting (E26) at n=0 into (E26) at n = 1, and then into (E26) at n = 2, … we obtain

$$\Delta\vec{u}_2 = A^2\Delta\vec{u}_0 + h\left(A\Delta\vec{\phi}_0 + \Delta\vec{\phi}_1\right) - \left(A\vec{e}_0(h) + \vec{e}_1(h)\right)$$

$$\Delta\vec{u}_{n+1} = A^{n+1}\Delta\vec{u}_0 + h\sum_{j=0}^{n} A^{n-j}\Delta\vec{\phi}_j - \sum_{j=0}^{n} A^{n-j}\,\vec{e}_n\left(h\right) \qquad \text{(E27)}$$

Taking norm of both sides, using $\left\|A^n\right\| \le C_A$ and the Lipschitz continuity $\left\|\Delta\vec{\phi}_n\right\| \le C_L\left\|\Delta\vec{u}_n\right\|$, we arrive at a recursive inequality for $\left\|\Delta\vec{u}_n\right\|$

$$\left\|\Delta\vec{u}_{n+1}\right\| \le C_A\left\|\Delta\vec{u}_0\right\| + hC_AC_L\sum_{j=0}^{n}\left\|\Delta\vec{u}_j\right\| + \left(n+1\right)C_AC_eh^{p+1} \qquad \text{(E28)}$$

To solve recursive inequality (E28), we introduce $\{g_n\}$ recursively as

$$g_0 = C_A\left\|\Delta\vec{u}_0\right\|$$

$$g_{n+1} = g_0 + hC_A C_L \sum_{j=0}^{n} g_j + \left(n+1\right) C_A C_e h^{p+1} \qquad \text{(E29)}$$

which ensures that

$$\left\|\Delta\vec{u}_n\right\| \le g_n \qquad \text{for all } n \ge 0$$

To calculate g_n, we re-write recursive equation (E29) as

$$g_{n+1} = \left(1 + hC_A C_L\right) g_n + C_A C_e h^{p+1}$$

$$\Longrightarrow \quad \left(1+hC_A C_L\right)^{-(n+1)} g_{n+1} \le \left(1+hC_A C_L\right)^{-n} g_n + \left(1+hC_A C_L\right)^{-(n+1)} C_A C_e h^{p+1}$$

$$\Longrightarrow \quad \left(1+hC_A C_L\right)^{-n} g_n \le g_0 + C_A C_e h^{p+1} \sum_{j=0}^{n-1} \left(1+hC_A C_L\right)^{-(n+1)}$$

$$\le g_0 + C_A C_e h^{p+1} \left(1+hC_A C_L\right)^{-1} \frac{1-\left(1+hC_A C_L\right)^{-n}}{1-\left(1+hC_A C_L\right)^{-1}}$$

$$\Longrightarrow \quad g_n \le \left(1+hC_A C_L\right)^{n} g_0 + C_A C_e h^{p+1} \frac{\left(1+hC_A C_L\right)^{n} - 1}{C_A C_L h}$$

$$\le \exp(nhC_A C_L) g_0 + \frac{\exp(nhC_A C_L) - 1}{C_L} C_e h^p$$

Using $\left\|\Delta\vec{u}_n\right\| \le g_n$ and $g_0 = C_A \left\|\Delta\vec{u}_0\right\|$, we obtain

$$\left\|\Delta\vec{u}_n\right\| \le C_A \exp\left(C_A C_L T\right) \left\|\Delta\vec{u}_0\right\| + \frac{\exp\left(C_A C_L T\right) - 1}{C_L} C_e h^p \qquad \text{for } nh \le T$$

When the initial value u_0 is at least p-th order accurate: $\left\|\Delta\vec{u}_0\right\| = \left\|\vec{u}_0 - \vec{v}_0\right\| \le c_{\text{int}} h^p$, the difference between the numerical solution u and exact solution v is bounded by

$$\left\|\vec{u}_n - \vec{v}_n\right\| \le \left(C_A \exp\left(C_A C_L T\right) c_{\text{int}} + \frac{\exp\left(C_A C_L T\right) - 1}{C_L} \right) h^p \qquad \text{for } nh \le T$$

which implies the LMM is convergent, by definition.