2 Survey of Numerical Methods for IVPs

2.1 Discrete Approximations:

Most numerical methods for IVPs determine a discrete approximation, $(x_i, y_i)_{i=0}^{N_{TOL}}$, to the true solution, $y(x)$, evaluated on a mesh, $a = x_0 < x_1 \cdots < x_{N_{TOL}} = b$. In general the mesh and N_{TOL} will depend on $f(x, y)$ and the specified accuracy tolerance, TOL (adaptive).

2.2 Accuracy and Error Control

1. The Global Error at x_i is $GE_i = y(x_i) - y_i$.
2. Numerical methods will generally determine y_{i+1} only after $(x_j, y_j)_{j=0}^{i}$ have been determined and a suitable stepsize, $h_{i+1} = (x_{i+1} - x_i)$, has been chosen. An underlying k-step formula is then applied,

$$y_{i+1} = y_i + h_{i+1}\Phi(x_i, y_{i+1}, y_i \cdots y_{i-k+1}).$$

   Note that:

   - The increment function, Φ, must depend on $f(\,)$.
   - Numerical methods are classified according to the value of k:

     $k = 1$ –One-step Methods (Runge-Kutta).

     $k > 1$ –Multistep Methods (Adams, BDF).
   - Numerical Methods can also be classified based on whether or not Φ depends on y_{i+1}. If there is a dependence the method is implicit, otherwise it is explicit.
3. The global error at x_{i+1} will then depend on GE_i, $y(x_i) - y_i$ as well as on h_{i+1} and Φ. In stepping to x_{i+1} the method cannot control $y(x_i) - y_i$, but h_{i+1} and possibly Φ can be adjusted to improve the accuracy.
4. Consider the 'Local' IVP associated with step i:

$$y' = f(x, y), \quad y(x_i) = y_i, \quad x \in [x_i, b].$$

Denote the true solution to this local problem, $z_i(x)$.

- If y_i were exact (ie. $y_i = y(x_i)$) then $y(x) \equiv z_i(x)$.
- When we apply a k-step formula, the best we can expect is that y_{i+1} will be a good approximation to $z_i(x_{i+1})$.
- **Definition:** The Local Error associated with step i, LE_i, is $z_i(x_{i+1}) - y_{i+1}$.
- The Global Error at x_{i+1} satisfies:

$$GE_{i+1} \equiv y(x_{i+1}) - y_{i+1} = \underbrace{(y(x_{i+1}) - z_i(x_{i+1}))}_{A} + \underbrace{(z_i(x_{i+1}) - y_{i+1})}_{B},$$

  where A depends on the global error at x_i and the 'mathematical stability' of the ODE (and is independent of the method) while B is the local error and depends only on h_{i+1} and Φ.
- **Definition:** Consider the application of a k-step formula to an IVP with a sufficiently differentiable $f(x,y)$. The formula is of order p iff for all sufficiently differentiable $z_i(x)$ we have (for constant h),

$$z(x_{i+1}) = z(x_i) + h\Phi(x_i, z(x_{i+1}) \cdots z(x_{i-k+1})) + O(h^{p+1}).$$

- Note that $z(x_{i+i}) - z(x_i) - h_{i+1}\Phi(x_i, z(x_{i+1}) \cdots z(x_{i-k+1}))$ is the Local Truncation Error, LTE, associated with the formula. For 1-step formulas one can show LTE = LE + $O(h^{p+2})$ but this is not the case for multistep formulas.

5. Classical Results for IVP methods:

   - For One-step methods, order p $\Leftrightarrow$ local error $= O(h^{p+1}) \Rightarrow$ global error $= O(h^p)$.
   - For multistep methods, order p plus numerical stability $\Leftrightarrow$ local error $= O(h^{p+1}) \Rightarrow$ global error $= O(h^p)$.

6. Numerical methods for IVPs often monitor the magnitude of the local error and 'accept' the step, (x_{i+1}, y_{i+1}) only if

$$\|est_i\| < TOL\, h_{i+1},$$

where TOL is a prescribed error tolerance and $est_i \approx z_i(x_{i+1}) - y_{i+1} = LE_i$. (Intuitively LE $= O(h\, TOL) \Rightarrow$ GE $= O(TOL)$.)

2.3 Alternative View of Error Control:

1. Often one can associate, with a discrete solution $(x_i, y_i)_{i=0}^{N_{TOL}}$, a 'natural' piecewise polynomial, $S(x)$, which interpolates the discrete solution and provides a continuous (possibly $C^1[a,b]$) approximation to $y(x)$. That is, for $x \in [x_i, x_{i+1}]$, $S(x)$ is a polynomial of degree $\leq k$, (which we will denote by $\hat{z}_i(x)$) interpolating the 'local data' y_i, y_{i+1}.

2. When such a computable interpolant, $S(x)$, is available then $\hat{z}_i(x) \approx z_i(x)$ over $[x_i, x_{i+1}]$. One can then ask how well $\hat{z}_i(x)$ satisfies the local IVP that defines $z_i(x)$. That is, one can define the approximate (or numerical solution) to be $S(x)$ and ask how well it satisfies the original IVP over [a, b].

3. The <u>defect</u> (or residual), $\delta(x)$, associated with this continuous numerical solution, $S(x)$, is defined for $x \in [a, b]$, to be

$$\delta(x) \equiv S'(x) - f(x, S(x)).$$

   Note:

   - $S(x)$ is computable and is a vector of piecewise polynomials (one for each component of $y(x)$).
   - $\delta(x)$ will be computable and continuous everywhere except possibly at the meshpoints, $x_1, x_2, \cdots x_{N_{TOL}-1}$.

4. The notion of GE and LE extend from the discrete to continuous case in a straightforward way,

$$CGE \equiv y(x) - S(x), \quad \text{for } \; x \in [a, b].$$

$$CLE \equiv z_i(x) - S(x) = z_i(x) - \hat{z}_i(x), \quad \text{for } \; x \in [x_i, x_{i+1}].$$

   The following theoretical results can be shown,

   - $\delta(x) = O(h^p) \Rightarrow CLE = O(h^{p+1}) \Rightarrow CGE = O(h^p)$.
   - If $\hat{z}_i(x)$ is computed on each step (and hence $S(x)$ determined globally), then the associated $\delta(x)$ can be sampled (at some extra cost) and a bound on $\|\delta(x)\|$ over (x_i, x_{i+1}) estimated. An error control strategy can 'accept' (x_{i+1}, y_{i+1}) only if this estimated bound is less than TOL. That is, an attempt is made to ensure $\|\delta(x)\| \leq TOL$ for all $x \in [a, b]$ and therefore $S(x)$ will satisfy exactly a slightly pertubed IVP, where the perturbation is bounded by TOL in norm.

2.4 Runge-Kutta methods:

Runge-Kutta methods are one-step methods where the ODE is sampled at s points in the interval $[x_i, x_{i+1}]$ and the associated $\Phi(x_i, y_i)$ is defined as a weighted average of these sampled values.

1. More specifically an s-stage Runge-Kutta formula uses s derivative evaluations and has the form:
$$y_{i+1} = y_i + h_{i+1}(\omega_1 k_1 + \omega_2 k_2 \cdots + \omega_s k_s),$$
where $h_{i+1} = x_{i+1} - x_i$ and, for $j = 1, 2 \cdots s$,
$$k_j = f(x_i + h_{i+1}\alpha_j, y_i + h_{i+1}\sum_{r=1}^{s} \beta_{jr} k_r).$$

2. This formula is characterized by the tableau,
$$\begin{array}{c|cccc} \alpha_1 & \beta_{11} & \beta_{12} & \dots & \beta_{1s} \\ \alpha_2 & \beta_{21} & \beta_{22} & \dots & \beta_{2s} \\ \vdots & \vdots & \vdots & & \vdots \\ \alpha_s & \beta_{s1} & \beta_{s2} & \dots & \beta_{ss} \\ \hline & \omega_1 & \omega_2 & \dots & \omega_s \end{array}$$
These $s^2 + 2s$ parameters are usually chosen to make implementation easier and/or to maximize the order. The maximum attainable order for an s-stage Runge-Kutta formula is $2s$.

3. The formula is <u>explicit</u> if $\beta_{jr} = 0$ for $r \geq j$. (Otherwise the equations defining the k_j's are a nonlinear system of sn equations in sn unknowns, which must be 'solved' on each step.) The maximum attainable order for an s-stage, explicit Runge-Kutta formula is given by the following table:

s	1	2	3	4	5	6	7	8	9	10	11
max order	1	2	3	4	4	5	6	6	7	7	8

4. An effective method must be able to estimate the local error (or the defect) on each step and adjust the stepsize to match the tolerance, TOL. As we have noted this strategy is an attempt to ensure that the maximum global error will be $O(TOL)$. For this reason Runge-Kutta <u>formula pairs</u> are widely used.

   An explicit, s-stage, order $(p, p+1)$ RK formula pair is defined in terms of two RK formulas (of orders p and $p+1$ respectively) that share the same k_j's. They can be characterized by the following tableau,
$$\begin{array}{c|ccccc} - & - & & & & \\ \alpha_2 & \beta_{21} & - & & & \\ \vdots & \vdots & & & & \\ \alpha_s & \beta_{s1} & \dots & & \beta_{s-1\,s} & - \\ \hline & \omega_1 & \omega_2 & \dots & & \omega_s \\ & \hat{\omega}_1 & \hat{\omega}_2 & \dots & & \hat{\omega}_s \end{array}$$

where

$$
\begin{aligned}
y_{i+1} &= y_i + h_{i+1} \sum_{r=1}^{s} \omega_r k_r \\
&= z_i(x_{i+1}) - c(f) h_{i+1}^{p+1} + O(h_{i+1}^{p+2}), \\
\hat{y}_{i+1} &= y_i + h_{i+1} \sum_{r=1}^{s} \hat{\omega}_r k_r \\
&= z_i(x_{i+1}) + O(h_{i+1}^{p+2}), \\
est_i &= (\hat{y}_{i+1} - y_{i+1}) \\
&= c(f) h_{i+1}^{p+1} + O(h_{i+1}^{p+2}).
\end{aligned}
$$

Note that the error estimate is a reliable estimate of the local error associated with the lower order (order p) formula. The following table gives the fewest number of stages required to define formula pairs of a given order.

order pair	(2,3)	(3,4)	(4,5)	(5,6)	(6,7)	(7, 8)
fewest stages	3	4	6	8	10	13

A straightforward implementation of a formula pair would then compute, on each attempted step, $\hat{y}_{i+1}$ and corresponding local error estimate, $est_i = y_{i+1} - \hat{y}_{i+1}$. The step would be accepted if $\|est_i\| < h_{i+1}\, TOL$ (such a heuristic is called error per unit step, EPUS). The heuristic for choosing the next trial stepsize would be to aim at achieving $\|est\| \approx h\, TOL$. For example, a typical heuristic is to use,

$$
h_{i+1} = .9 \left(\frac{TOL\ h_i}{\|est_i\|} \right)^{1/p} h_i,
$$

where .9 is a 'safety factor'. This formula works for use after a rejected step as well but must be modified slightly when round-off errors are significant.

5. Interpolants for RK formula pairs (ie. defining an appropriate $S(x)$):

   For an s-stage, order $(p, p+1)$ RK formula pair one can determine suitable computable interpolants, $S(x)$, on the mesh $a = x_0 < x_1 \cdots < x_{N_{TOL}} = b$ (associated with the underlying discrete method). There are several alternatives possible and we will consider one of the least expensive alternatives. There is a trade-off that arises between the cost and reliabilty of the resulting interpolants.

   A p^{th}-order, s-stage RK formula determines

$$
y_{i+1} = y_i + h_{i+1} \sum_{j=1}^{s} \omega_j k_j,
$$

where

$$
k_j = f(x_i + h_{i+1}\alpha_j, y_i + h_{i+1} \sum_{r=1}^{s} \beta_{jr} k_r).
$$

A Continuous extension (CRK) is determined by adding $(\bar{s} - s)$ extra stages to obtain an order p approximation for $x \in (x_i, x_{i+1})$

$$u_i(x) = y_i + h_{i+1} \sum_{j=1}^{\bar{s}} b_j\left(\frac{x - x_i}{h_{i+1}}\right) k_j,$$

where $b_j(\tau)$ is a polynomial of degree p and $\tau = \frac{x - x_i}{h_{i+1}}$.

- The $[u_i(x)]_{i=1}^{N_{TOL}}$ define the piecewise polynomial $S(x)$ for $x \in [a, b]$. This will be considered the numerical solution generated by the CRK method.
- $S(x) \in C^0[a, b]$ and will interpolate the underlying discrete RK values, y_i, if $b_j(1) = \omega_j$ for $j = 1, 2 \cdots s$ and $b_{s+1}(1) = b_{s+2}(1) = \cdots b_{\bar{s}}(1) = 0$.
- Similarly a simple set of constraints on the $b_j'(\tau)$, will ensure $S'(x)$ interpolates $f(x_i, y_i)$ and therefore $S(x) \in C^1[a, b]$.

Formula	p	s	$\bar{s}$
CRK4	4	4	6
CRK5	5	6	9
CVSS6B	6	7	11
CVSS7	7	9	15
CVSS8	8	13	21

Table 1: Cost per step of some CRK formulas

2.5 A Numerical Example:

A predator-prey relationship can be modeled by the IVP:

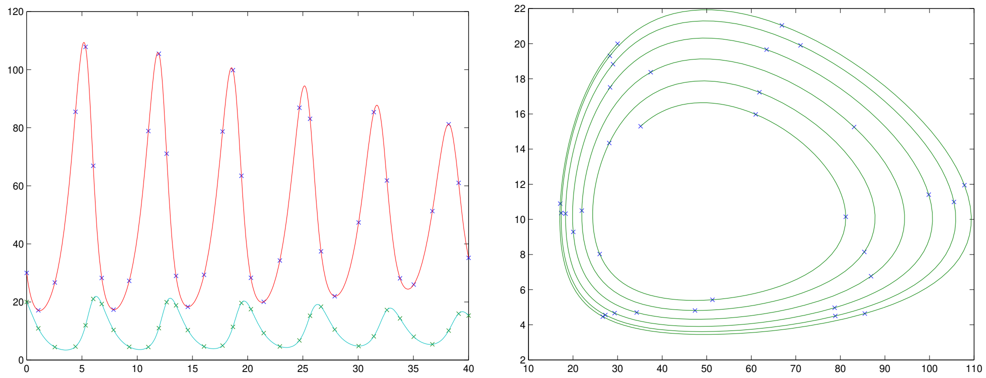
$$y_1' = y_1 - 0.1 y_1 y_2 + 0.02x$$
$$y_2' = -y_2 + 0.02 y_1 y_2 + 0.008x$$

with

$$y_1(0) = 30, \quad y_2(0) = 20.$$

Here $y_1(x)$ represents the 'prey' population at time x and $y_2(x)$ represents the 'predator' population at time x. The solution can then be visualized as a standard x/y solution plot or by a 'phase plane' plot. Figure 1 illustrates the solution to this system. We know that for different initial conditions solutions to this problem exhibit oscillatory behaviour as x increases.

A biologist may be interested in whether the solutions to this equation are 'almost periodic' (in the sense that the difference between successive maximum is constant) and whether the local maxima approach a steady state exponentially. (See Figure 2).



(a) Solution Plot (b) Phase Plane Plot

Figure 1: Solutions to the predator prey problem for $x \in [0, 40]$

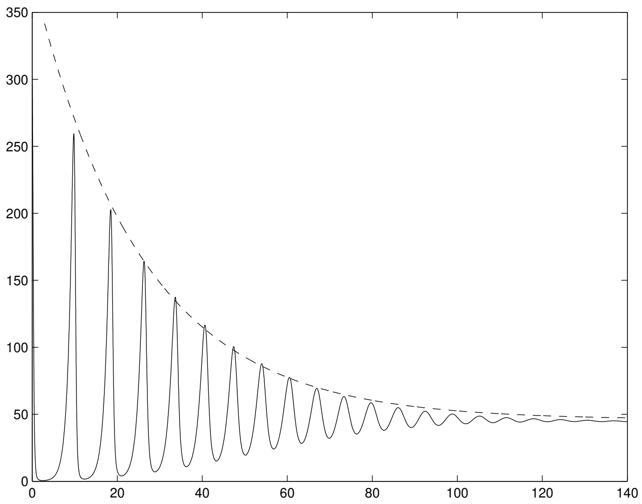


Figure 2: Typical behaviour of prey or predator population and decay to steady state

2.6 Multistep Methods:

1. A Linear Multistep Formula (LMF) has the form,

$$y_{i+1} = \sum_{j=1}^{k} \alpha_j y_{i+1-j} + h \sum_{j=0}^{k} \beta_j y'_{i+1-j}.$$

   This formula is implicit if $\beta_0 \neq 0$ and implicit formulas are usually preferred because of their superior stability and accuracy properties. For implicit formulas we have the RHS of this equation contains the term, $\beta_0 y'_{i+1} = \beta_0 f(x_{i+1}, y_{i+1})$, and therefore a nonlinear system of equations must be solved on each step.

2. To solve the nonlinear systems of equations, a Predictor-Corrector (P-C) iteration

is usually used,

$$\text{P:}\;\; y^0_{i+1} = \sum_{j=1}^{k} \bar{\alpha}_j y_{i+1-j} + h \sum_{j=1}^{k} \bar{\beta}_j y'_{i+1-j} \;\; \text{an explicit formula}$$

Iterate or 'Correct' for $r = 0, 1 \cdots M$:

$$\begin{aligned} \text{E:}\;\; y'^{(r)}_{i+1} &= f(x_{i+1}, y^r_{i+1}), \;\; \text{'Evaluate'} \\ \text{C:}\;\; y^{r+1}_{i+1} &= \sum_{j=1}^{k} \alpha_j y_{i+1-j} + h \sum_{j=1}^{k} \beta_j y'_{i+1-j} + h\beta_0 y'^{(r)}_{i+1}, \;\; \text{'Correct'} \end{aligned}$$

- This iteration is denoted $P(EC)^M$ or $P(EC)^M E$.
- Convergence is guaranteed provided $\|h\frac{\partial f}{\partial y}\| < \frac{1}{|\beta_0|}$.
- $M = 1$ or $M = 2$ has been found to be optimal for most standard problems.

3. Classical Results for uniform mesh (constant stepsize) Multistep Methods:
   - A LMF is <u>Stable</u> (as $h \to 0$) iff the associated characteristic polynomial, $\alpha(z) = z^k - \sum_{j=1}^{k} \alpha_j z^{k-j}$ has all its roots inside or on the unit circle.
   - A LMF is <u>Consistent</u> if $\alpha(1) = 0$ and $\alpha'(1) = \beta(1)$ (where $\beta(z) \equiv \sum_{j=0}^{k} \beta_j z^{k-j}$).
   - A LMF is <u>Convergent</u> if the maximum Global Error $\to 0$ as $h \to 0$.
   - Key Theorem:– A LMF is Convergent iff it is Consistent and Stable.
   - Order of Convegence for a LMF:
     (a) The <u>Local Truncation Error</u> (LTE) for a LMF is defined, for sufficiently differentiable $y(x)$, to be,

     $$LTE \equiv L(h, y(x)) \equiv y(x_i+h) - \sum_{j=1}^{k} \alpha_j y(x_i+(1-j)h) - h\sum_{j=0}^{k} \beta_j y'(x_i+(1-j)h).$$

     (b) Recall: A formula is of order p iff $LTE = O(h^{p+1})$.
     (c) **Key Result:** Stability + $LTE = O(h^{p+1}) \Leftrightarrow LE = O(h^{p+1}) \Rightarrow y(x_i) - y_i = O(h^p)$.

4. Derivation of high order LMFs:
   - One can expand $L(h, y(x))$ in a power series in h and choose the coefficients $\alpha_1, \alpha_2 \cdots \alpha_k, \beta_0, \beta_1 \cdots \beta_k$ to satisfy $LTE = O(h^{p+1})$, for as large a p as possible, subject to the stability constraint.
   - Dahlquist's Key Result (1956):
     (a) The maximum order of a k-step, stable LMF is $k + 1$.

(b) It is always possible to arbitrarily prescribe the $\alpha_1, \alpha_2 \cdots \alpha_k$ to satisfy stability and then determine $\beta_0, \beta_1 \cdots \beta_k$ to obtain order $k+1$.

(c) The obvious choice of $\alpha(z) = z^k - z^{k-1}$, leads to Adams Formulas.

5. Special Class of LMF: Adams Formulas.

Adams Moulton Corrector formula of order $k+1$:

$$y_{i+1} = y_i + h \sum_{j=0}^{k} \beta_j y'_{i+1-j}.$$

Adams Bashforth Predictor formula of order k,

$$y_{i+1} = y_i + h \sum_{j=1}^{k} \bar{\beta}_j y'_{i+1-j}.$$

These are the most widely used LMFs for standard IVPs.

6. Extension of Adams LMFs to allow variable stepsize:

- One can write the local solution to an IVP evaluated at $x_i + h,\ z_i(x_i + h)$, as

$$z_i(x_i + h) = y_i + \int_{x_i}^{x_i+h} f(s, z_i(s))ds.$$

Adams formulas can be interpreted as being based on approximating the integral on the RHS of this expression by the integral of a polynomial that interpolates $f(s, z_i(s))$ at $x_i + (1-j)h,\ \ (j = 0, 1 \cdots k)$ using $y_{i+1-j} \approx z_i(x_{i+1-j})$. This interpretation yields a closed-form expression for the constant stepsize coefficients, $\beta_r, \bar{\beta}_r$ for the k-step AM and AB formulas. It also suggests a natural extension of these formulas to the case of variable stepsize.

- Let $\bar{P}_{i,k}(s; y'_i, y'_{i-1} \cdots y'_{i-k})$ be the unique polynomial (in s) of degree $\leq k$ satisfying,

$$\bar{P}_{i,k}(x_{i+1-j}; y'_i, y'_{i-1} \cdots y'_{i-k}) = y'_{i+1-j},\ j = 1, 2 \cdots (k+1),$$

where the x_i's need not be equally spaced. Similarly let $P_{i,k}(s; y'_{i+1}, y'_i \cdots y'_{i-k})$ be the unique polynomial of degree $\leq k+1$ satisfying,

$$P_{i,k}(x_{i+1-j}; y'_{i+1}, y'_i \cdots y'_{i-k}) = y'_{i+1-j},\ j = 0, 1 \cdots (k+1).$$

- An Efficient $P(EC)^M$ variable step Adams Formula can be implemented as:

$$\text{P:}\ \ y^0_{i+1} = y_i + \int_{x_i}^{x_i+h} \bar{P}_{i,k}(s; y'_i, y'_{i-1} \cdots y'_{i-k})ds;$$

For $r = 0, 1 \cdots M$:

$$\begin{aligned} \text{E:}\ \ y'^{(r)}_{i+1} &= f(x_{i+1}, y^r_{i+1}), \\ \text{C:}\ \ y^{r+1}_{i+1} &= y_i + \int_{x_i}^{x_i+h} P_{i,k}(s; y'^{(r)}_{i+1}, y'_i \cdots y'_{i-k})ds; \end{aligned}$$

Note that this iteration will converge ($r \to \infty$) if $\|h\frac{\partial f}{\partial y}\| < \frac{1}{|\beta_0|}$. A Newton basis or Lagrange basis can be used effectively for implementation.

- For Adams methods a natural definition for an accurate piecewise polynomial, $S(x)$, valid for all $x \in [a, b]$ is,

$$S(x) \equiv y_i + \int_{x_i}^{x} P_{i,k}(s; y'_{i+1}, y'_i \cdots y'_{i-k})ds \;\text{ for }\; x \in [x_i, x_{i+1}].$$

- Error estimates and local extrapolation:

We have for $M > 1$,

$$\begin{aligned} z_i(x_i + h) - y^0_{i+1} &= O(h^{k+1}), \\ z_i(x_i + h) - y^M_{i+1} &= O(h^{k+2}). \end{aligned}$$

Therefore $y^M_{i+1} - y^0_{i+1}$ is a valid estimate for the local error of y^0_{i+1} and we can can control the local error in y^M_{i+1} <u>indirectly</u> by keeping,

$$\|est_i\| = \|y^M_{i+1} - y^0_{i+1}\| < TOL.$$

Virtually all variable step, variable order Adams methods are implemented this way. This includes the methods: GEAR, LSODE, VODE, DE/STEP, DVDQ, DEABM, IVPAG and D02CHF.

2.7 Stiff ODEs:

1. Motivation: – Mathematical simulations of systems that have different components evolving on different time scales often lead to ODEs which exhibit numerical difficulties which require special attention and special methods. The associated class of IVPs are called stiff IVPs and we will present an overview of the cause of the difficulty and software that is available for this class of problems.

   - Consider the model scalar IVP:

   $$y' = \lambda y, \;\; y(x_0) = y_0, \;\; \lambda \in \Re, \;\; \lambda << 0.$$

   The solution is,

   $$y(x) = e^{\lambda(x-x_0)}y_0, \;\text{ on any interval }\; [x_0, x_F],$$

   and therefore $|y(x)| = e^{\lambda(x-x_0)}|y_0|$.

Let $\hat{x} = x_0 + \ln(\frac{TOL}{|y_0|})/\lambda$ (ie. $|y(\hat{x})| = TOL$). For this model problem we then have,

(a) For $x \in [x_0, \hat{x}]$, $|y(x)| > TOL$ and we require and expect the numerical solution to remain accurate.

(b) For $x > \hat{x}$, $|y(x)| < TOL$, accuracy is no longer important as long as the numerical solution remains stable (ie. $|S(x)| < TOL$).

(c) $[x_0, \hat{x}]$ is the transient region and $[\hat{x}, x_F]$ is the steady state (or 'smooth') region.

- Consider applying Euler's formula to this problem (simplest one-step formula),

$$y_{i+1} = y_i + hy_i' = (1 + h\lambda)y_i,$$

When $|\lambda|$ is large, $h\lambda << -2$, $|1 + h\lambda| >> 1$ and,

$$|y_{i+1}| = |1 + h\lambda|^{i+1}|y_0|.$$

This leads to an unstable computation.

- Consider applying the Implicit Euler (Backward Euler) formula to this problem,

$$y_{i+1} = y_i + hy_{i+1}' = \frac{1}{(1 - h\lambda)}y_i,$$

When $|\lambda|$ is large, $h\lambda << -2$, $|\frac{1}{(1-h\lambda)}| << 1$ and,

$$|y_{i+1}| = |\frac{1}{(1 - h\lambda)}|^{i+1}|y_0|.$$

This leads to a very stable computation as the $|y_i| \to 0$ rapidly.

2. **Fixed h stability:** Consider the model problem,

$$y' = \lambda y, \;\; y(x_0) = y_0, \;\; \text{where } \lambda \in C.$$

- A formula, when applied to this problem with constant stepsize h, is considered stable (for the particular values of h and λ) if $|y_i|$ remains bounded as $i \to \infty$ for any initial condition vector, y_0.
  (a) When an explicit, s-stage, Runge-Kutta formula is applied to this model problem, the numerical solution satisfies,

  $$y_{i+1} = p_s(h\lambda)y_i \;=\; [p_s(h\lambda)]^{i+1} y_0,$$

  where $p_s(w)$ is a polynomial of degree $\leq s$ whose coefficients are determined by the parameters that define the Runge-Kutta formula.

  (b) When a k-step LMF is applied to this model problem, the numerical solution satisfies,

  $$y_i = \sum_{j=1}^{k} c_j \xi_j^i,$$

  where the c_j's depend on the starting values,$y_0, y_1 \cdots y_{k-1}$ and the ξ_j's are the roots of the associated characteristic polynomial, $\alpha(w) - h\lambda\beta(w)$. In this case $|y_i|$ is bounded iff $|\xi_j| \leq 1, \; j = 1, 2 \cdots k.$ (Note that $\alpha(w)$ has all roots inside or on the unit circle and this implies as $h\lambda \to 0$, the formula will be stable.)

- **Definition**
  The Region of absolute stability associated with a formula is

  $$\mathcal{R} \equiv \{h\lambda : \text{when applied to the model problem with constant } h, |y_i| \to 0\}.$$

  Therefore for an explicit Runge-Kutta formula we have,

  $$\mathcal{R} = \{h\lambda : |p_s(h\lambda)| \leq 1\},$$

  and for a multistep formula we have,

  $$\mathcal{R} = \{h\lambda : \text{ all roots of } \;\alpha(w) - (h\lambda)\beta(w)\; \text{ are inside the unit circle}\}.$$

3. In determining the step from x_i to $x_i + h$ three factors can impose restrictions on h:

   - The stability of the numerical solution ($h\lambda \in \mathcal{R}$).
   - The convergence of the Predictor-Corrector iteration,

   $$h|\frac{\partial f}{\partial y}| < \frac{1}{|\beta_0|} \text{ or } (h\lambda < \frac{1}{|\beta_0|}).$$

   - The control of the local error ($|C(f)|h^{p+1} < TOL\ h$).

   Definition: AN ODE will be considered Stiff if the stepsize used during integration by conventional methods is determined by stability or convergence of an iteration, rather than by control of the local error.

   Note that by conventional methods we mean standard Adams multistep or standard explicit Runge-Kutta methods. This 'definition' implies that whether a problem is stiff or not will depend on TOL and the length of integration $(x_F - x_0)$ and a problem can be stiff for part of the integration.

4. Consider the Example,

   $$y' = \begin{bmatrix} -1 & 0 \\ 0 & -100 \end{bmatrix} y, \ \ y(0) = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$

   The solution to this IVP is,

   $$y(x) = \begin{bmatrix} e^{-x} \\ e^{-100x} \end{bmatrix}.$$

   - The second component is a transient, and is only significant in an initial interval as it decays rapidly to zero. The accuracy of this component is only of concern in this initial transient region after which only the stability of this component and the accuracy of the first component is required.
   - Conventional methods will require that $|h100|$ remain small throughout the integration ($|h\lambda| < \frac{1}{|\beta_0|}$ or $h\lambda \in \mathcal{R}$).
   - In general, for a system of ODEs, $y' = Ay$, numerical stability requires that $\forall \lambda_j \in \rho(A),\ h\lambda_j \in \mathcal{R}$.

5. For the model problem, $y' = \lambda y$ we know that $|y(x_i)|$ remains bounded iff $Re(\lambda) \leq 0$. Therefore an ideal stability situation would be to develop formulas whose regions of absolute stability include the LHP.

   - Definition (Dahlquist): A formula is A-Stable if its region of absolute stability, $\mathcal{R}$ contains the LHP, $\{h\lambda : Re(h\lambda) < 0\}$. An example of an A-Stable formula is the implicit Euler formula.
   - Key Result (Dahlquist 1963): The maximum order of an A-stable LMF is 2.

- A formula is A(α)-Stable if its stability region contains the wedge-shaped region, S(α),
$$S(\alpha) = \{z : \arg(-z) < \alpha \leq \pi/2\}.$$
Clearly a formula is A-Stable iff it is A($\pi/2$)-Stable.
- For $\alpha > 0$, a formula (multistep or one-step) can be A(α)-Stable only if it is implicit.

6. **Suitable Methods for Stiff IVPs:**

Implicit Runge-Kutta methods For a general, s-stage IRK formula applied to $y' = \lambda y$ we have,
$$y_{i+1} = \left[\frac{P_s(h\lambda)}{Q_s(h\lambda)}\right] y_i,$$
where $P_s(w)$ and $Q_s(w)$ are polynomials of degree $\leq s$. This implies,
$$\mathcal{R} = \{z : |\frac{P_s(z)}{Q_s(z)}| \leq 1\}.$$
Therefore it would be appropriate to derive Runge-Kutta formulas whose corresponding rational functions approximate e^z well for $z \in$ LHP. High order, A-Stable, IRK methods are available but tend to be expensive to use because of the nonlinear systems of equations that must be solved on each time step. This has been done and there are several packages available for stiff problems based on such formulas.

Multistep Methods – BDF formulas (Gear 1967)
Recall that for a k-step LMF we have,
$$\mathcal{R} = \{z : \alpha(w) - z\beta(w) \text{ has all its roots inside U.C. }\}.$$
If we want A(α)-Stability the formula must be implicit (ie. $\beta(w)$ is of degree k). In addition, all the roots of $\beta(w)$ must be inside or on the unit circle. Gear suggested setting $\beta(w) = \beta_0 w^k$, and then set the $\alpha_j,\ j = 1, 2 \cdots k$ and β_0 to obtain as high an order as possible:
$$y_{i+1} = \sum_{j=1}^{k} \alpha_j y_{i+1-j} + h\beta_0 y'_{i+1}.$$
Note that with this approach,

(a) We can derive k-step formulas of order k for all $k > 1$.
(b) These formulas are only stable ($h \to 0$) for $k \leq 6$. The formula with $k = 6$ is generally not recommended because of its small region of absolute stability.
(c) This class of multistep formulas are called Backward Differentiation Formulas (BDF) and form the basis for the most widely used class of methods for Stiff problems.

(d) The BDF formulas can be extended to variable stepsize in a natural way as follows.
Let $Q_{i,k}(s; y_{i+1}, y_i \cdots y_{i+1-k})$ be the unique polynomial (in s) of degree $\leq k$ that satisfies,

$$Q_{i,k}(x_{i+1-j}; y_{i+1}, y_i \cdots y_{i+1-k}) = y_{i+1-j}, \quad j = 0, 1 \cdots k.$$

We then can 'define' y_{i+1} to be the unique solution of the nonlinear equation (in y),

$$Q'_{i,k}(x_{i+1}; y, y_i \cdots y_{i+1-k}) = f(x_{i+1}, Q_{i,k}(x_{i+1}; y, y_i \cdots y_{i+1-k})).$$

This observation leads to a closed form expression for β_0 and the α_j's that define the BDF formula for any k and any $h > 0$. It also suggests a natural piecewise polynomial, $S(x)$, that will give an accurate approximation to $y(x)$ for any $x \in [a, b]$. That is,

$$S(x) \equiv Q_{i,k}(x; y_{i+1}, y_i \cdots y_{i+1-k}), \quad \text{for } x \in [x_i, x_{i+1}].$$

3 Initial-Value based BVP Methods

3.1 Superposition (for Linear Problems)

This section is based on a sequence of Technical Reports and publications by M. Scott, H.A. Watts and colleagues who implemented a family of Two-point BV Codes at SANDIA Laboratories in the period 1978 - 1980. Extensions of the basic method, SUPORT, described in this section to handle nonlinear problems (SUPORQ) and eigenvalue problems (SUPORE) are also available.

1. Consider the Linear BVP,
$$y' = F(x)y + g(x),$$
with separated boundary conditions,
$$Ay(a) = y_a, \quad By(b) = y_b,$$
where $y, \ g \in \Re^n$, F is an $n \times n$ matrix, A is an $(n-k) \times n$ matrix of rank $n-k$, B is a $k \times n$ matrix of rank k, $y_a \in \Re^{n-k}$ and $y_b \in \Re^k$.

2. The Principle of Superposition (Mathematical Justification)

   - let $u_1(x), u_2(x) \cdots u_k(x)$ be <u>linearly independent</u> solutions of,
   $$u' = F(x)u,$$
   with initial conditions chosen so that
   $$Au_i(a) = 0, \quad \text{for } i = 1, 2 \cdots k.$$
   (That is, the $u_i(a)$'s span the null space of A.) Furthermore, let $v(x)$ satisfy,
   $$v' = F(x)v + g(x), \quad Av(a) = y_a.$$