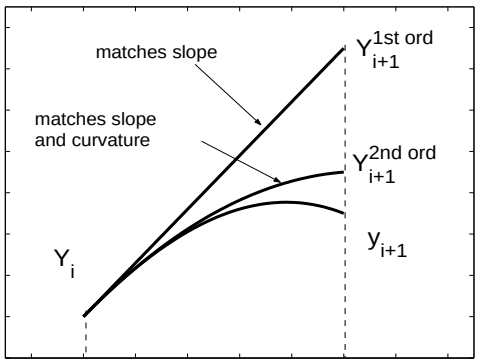
3 Multistep, Predictor-Corrector, and Implicit methods

In this section, we will introduce methods that may be as accurate as high-order Runge-Kutta methods but will require fewer function evaluations.

We will also introduce implicit methods, whose significance will become clearer in a later section.

3.1 Idea behind multistep methods

The figure on the right illustrates the (familiar) fact that if you know $y'(x_i)$, i.e. the *slope* of $y(x)$, then you can compute a first-order accurate approximation $Y_{i+1}^{\text{1st order}}$ to the solution y_{i+1}.
Likewise, if you know the slope and the *curvature* of your solution at a given point, you can compute a second-order accurate approximation, $Y_{i+1}^{\text{2nd order}}$, to the solution at the next step.



Now, recall that curvature is proportional to y''. This motivates the following.

Question: How can we find approximation to y_i'' using already computed values Y_{i-k}, $k = 0, 1, 2, \ldots$?

Answer: Note that

$$y_i'' \approx \frac{y_i' - y_{i-1}'}{h} = \frac{f_i - f_{i-1}}{h}\,. \tag{3.1}$$

Here and below we will use the notation f_i in two slightly different ways:

$$f_i \equiv f(x_i, y_i) \qquad \text{or} \qquad f_i \equiv f(x_i, Y_i) \tag{3.2}$$

whenever this does not cause any confusion.

Continuing with Eq. (3.1), we can state it more specifically by writing

$$y_i'' = \frac{y_i' - y_{i-1}'}{h} + O(h) = \frac{f_i - f_{i-1}}{h} + O(h)\,, \tag{3.3}$$

where we will compute the $O(h)$ term later. For now, we use (3.3) to approximate y_{i+1} as follows:

$$\begin{aligned} y_{i+1} = y(x_i + h) = & \quad y_i + h y_i' + \frac{h^2}{2} y_i'' + O(h^3) \\ = & \quad y_i + h f_i + \frac{h^2}{2}\left(\frac{f_i - f_{i-1}}{h} + O(h)\right) + O(h^3) \\ = & \quad y_i + h\left(\frac{3}{2} f_i - \frac{1}{2} f_{i-1}\right) + O(h^3)\,. \end{aligned} \tag{3.4}$$

Remark 1: To start the corresponding finite-difference method, i.e.

$$Y_{i+1} = Y_i + h\left(\frac{3}{2} f_i - \frac{1}{2} f_{i-1}\right) \tag{3.5}$$

(now we use f_i as $f(x_i, Y_i)$), one needs *two* initial points of the solution, Y_0 and Y_1. These can be computed, e.g., by the simple Euler method; this is discussed in more detail in Section 3.4.

Remark 2: Equation (3.4) becomes *exact* rather than approximate if $y(x) = p_2(x) \equiv ax^2 + bx + c$ is a second-degree polynomial in x. Indeed, in such a case,

$$y_i' = 2ax_i + b, \qquad \text{and} \qquad y_i'' = 2a = \frac{y_i' - y_{i-1}'}{h}\,; \tag{3.6}$$

(note the exact equality in the last formula). We will use this remark later on.

Method (3.5) is of the second order. If we want to obtain a third-order method along the same lines, we need to use the third derivative of the solution:

$$y_i''' = \frac{y_i' - 2y_{i-1}' + y_{i-2}'}{h^2} + O(h) \tag{3.7}$$

(you will be asked to verify this equation in one of the homework problems). Then we proceed as in Eq. (3.4), namely:

$$y_{i+1} = y_i + hy_i' + \frac{h^2}{2}y_i'' + \frac{h^3}{6}y_i''' + O(h^4)\,. \tag{3.8}$$

If you now try to substitute the expression on the r.h.s. of (3.3) for y_i'', you will notice that due to the $O(h)$-term in (3.3), the overall error in (3.8) would become $O(h^3)$, whereas you want to have the error $O(h^4)$ for a 3rd-order method. To address that, let us modify (3.3) so that the error there becomes $O(h^2)$. Here is the corresponding calculation:

$$\begin{aligned} \frac{y_i' - y_{i-1}'}{h} &= \frac{y'(x_i) - y'(x_{i-1})}{h} \\ &\overset{\text{Taylor for } y(x_i - h)}{=} \frac{y_i' - \left[y_i' - hy_i'' + \frac{h^2}{2}y_i''' + O(h^3)\right]}{h} \\ &= y_i'' - \frac{h}{2}y_i''' + O(h^2), \end{aligned} \tag{3.9}$$

whence

$$y_i'' = \frac{y_i' - y_{i-1}'}{h} + \frac{h}{2}y_i''' + O(h^2)\,. \tag{3.10}$$

To complete the derivation of the third-order finite-difference method, we substitute Eqs. (3.10), (3.7), and $y_i' = f_i$ etc. into Eq. (3.8). The result is:

$$Y_{i+1} = Y_i + \frac{h}{12}[23f_i - 16f_{i-1} + 5f_{i-2}]\,; \tag{3.11}$$

the local truncation error of this method is $O(h^4)$. Method (3.11) is called the 3rd-order *Adams–Bashforth* method.

Similarly, one can derive higher-order Adams–Bashforth methods. For example, the 4th-order Adams–Bashforth method is

$$Y_{i+1} = Y_i + \frac{h}{24}[55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3}]\,. \tag{3.12}$$

Methods like (3.5), (3.11), and (3.12) are called *multistep* methods. To start a multistep method, one requires more than one initial point of the solution (in the examples considered above, *the number of required initial points equals the order of the method*).

In Appendix 2 we present an alternative — matrix — form, in which one can cast the Adams–Bashforth and even more general multistep methods given by Eq. (3.17) below.

Comparison of Adams–Bashforth multistep and Runge–Kutta methods

The advantage of multistep over single-step RK methods of the same accuracy is that the multistep methods require only one function evaluation per step, while, e.g., the cRK method requires 4, and the RK–Fehlberg method, 6, function evaluations.

The disadvantages of *Adams–Bashforth*[5] multistep methods are primarily related to the difficulties that one would face by making them use a varying step size. In contrast, for single-step RK methods, this is a straightforward procedure, as we learned in Lecture 2. Below I list some of those difficulties for multistep methods.

1. Using a variable step size requires either an interpolation of the numerical solution from a variable- onto a constant-step size grid, or using rather complicated coefficients involving ratios of step sizes, say h_i/h_{i-1}, instead of the constants like 3/2, 1/2 in formulae (3.5) etc.
2. If function $f(x,y)$ changes abruptly at $x = x_{\text{change}}$, as in the problems in Homework # 2 involving an opening parachute, then a RK method would require only one step to adjust to the new values of f beyond x_{change}. In contrast, a k-step method would take k steps to "recover". That is, over those k steps taken past the point x_{change} it would be using the information about the "pre-x_{change}" values of f, which are irrelevant for what is happening beyond x_{change}.[6]
3. In Sec. 3.6 we will learn about error control in so-called predictor–corrector methods (see Sec. 3.5), which is based on a key fact examplified by Eqs. (3.35), (3.36), and the pair of equations below them. Namely, the local truncation errors of the predictor and corrector equations must be proportional to each other. However, the derivation of this fact, part of which is given in Sec. 3.8, hinges on the assumption that all steps have equal size.

Despite the aforementioned difficulties, multistep methods with a variable step size have been (and still are) proposed and studied, but they are not of the Adams–Bashforth type. They are based on the idea by Arnold Nordsieck published in 1962; his paper is posted next to this Lecture.[7] The idea of the Nordsieck methods is described in Appendix 3. Practical codes based on them have been constructed (primarily at the Lawrence Livermore National Laboratory); they were originally called LSODE and VODE, but later these names evolved. A very brief overview of them can be found in `http://lh3lh3.users.sourceforge.net/solveode.shtml`.

3.2 An alternative way to derive formulae for multistep methods

Recall that the 2nd-order Adams–Bashforth method (3.5) was *exact* on solutions $y(x)$ that are 2nd-degree polynomials: $y(x) = p_2(x)$ (see Remark 2 after Eq. (3.4)). Similarly, one expects that the 3rd-order Adams–Bashforth method should be exact for $y(x) = p_3(x)$. We will now use this observation to derive the formula for this method, Eq. (3.11), in a different manner than in Sec. 3.1.

To begin, we take, according to the above observation, $f(x,y) = y'(x) = (p_3(x))' = p_2(x)$, i.e. a 2nd-degree polynomial in x. We now integrate the differential equation $y' = f(x,y)$ from x_i to x_{i+1} and obtain:

$$y_{i+1} = y_i + \int_{x_i}^{x_{i+1}} f(x, y(x))dx\,. \tag{3.13}$$

[5] Note that this pertains only to *this* type of multistep methods and to their generalization (3.17) considered later in this Lecture. This disadvantage can be overcome for certain other multistep methods, as noted below.

[6] This reason was suggested by Mr. Jacob Wahlen-Strothman, who took this course in 2012.

[7] It is well written, but is not an easy reading.

Let us approximate the integral by a quadrature formula, as follows:

$$\int_{x_i}^{x_{i+1}} f(x,y(x))dx \approx h(b_0 f_i + b_1 f_{i-1} + b_2 f_{i-2}) \tag{3.14}$$

and *require* that the above equation hold *exactly*, rather than approximately, for any $f(x,y(x)) = p_2(x)$. This is equivalent to requiring that (3.14) hold exactly for $f = 1$, $f = x$, and $f = x^2$. Without loss of generality,[8] one can set $x_i = 0$ and then rewrite Eq. (3.14) for the above three forms of f:

$$\begin{aligned}
&\text{for } f=1: \quad \int_0^h 1\,dx \;=\; h \;=\; h\,(\,b_0\cdot 1 + b_1\cdot 1 + b_2\cdot 1\,)\\
&\text{for } f=x: \quad \int_0^h x\,dx \;=\; \tfrac{1}{2}h^2 \;=\; h\,(\,b_0\cdot 0 + b_1\cdot(-h) + b_2\cdot(-2h)\,)\\
&\text{for } f=x^2: \quad \int_0^h x^2\,dx \;=\; \tfrac{1}{3}h^3 \;=\; h\,(\,b_0\cdot 0 + b_1\cdot(-h)^2 + b_2\cdot(-2h)^2\,)\,.
\end{aligned} \tag{3.15}$$

Equations (3.15) constitute a linear system of 3 equations for 3 unknowns b_0, b_1, and b_2. Solving it, we obtain

$$b_0 = \frac{23}{12}, \qquad b_1 = -\frac{16}{12}, \qquad b_2 = \frac{5}{12},$$

which in combination with Eq. (3.14) yields the same method as (3.11). Methods of higher order can be obtained similarly.

3.3 A more general form of multistep methods, with examples

The Adams–Bashforth methods above have the following common form:

$$Y_{i+1} - Y_i = h\sum_{k=0}^{N} b_k f_{i-k}\,. \tag{3.16}$$

As has been shown in Sec. 3.2, the sum on the r.h.s. approximates

$$\int_{x_i}^{x_{i+1}} f(x,y(x))dx\,.$$

Let us now consider multistep methods of a more general form:

$$Y_{i+1} - \sum_{k=0}^{M} a_k Y_{i-k} = h\sum_{k=0}^{N} b_k f_{i-k}\,, \tag{3.17a}$$

where

$$\sum_{k=0}^{M} a_k = 1. \tag{3.17b}$$

The last condition intuitively makes sense since the sum on the l.h.s. of (3.17a) replaces the term $1\cdot Y_i$ in (3.16). Note that if we rewrite the l.h.s. of (3.17a) as

$$\sum_{k=0}^{M} a_k (Y_{i+1} - Y_{i-k}),$$

[8] In a homework problem, you will be asked to show this.

where we have used (3.17b), then the r.h.s. of (3.17a) acquires an interpretation that is similar to that of the r.h.s. of (3.16). Namely, it approximates

$$\sum_{k=0}^{M} a_k \int_{x_{i-k}}^{x_{i+1}} f(x, y(x))dx\,.$$

In the next Lecture, we will discover that many methods of the form (3.17) have a serious flaw in them, but for now let us consider two particular examples, focusing only on the accuracy of the following methods.

Simple center-difference (Leap-frog) method

Recall that

$$\frac{y_i - y_{i-1}}{h} = y_i' + O(h)\,. \tag{3.18}$$

However,[9]

$$\frac{y_{i+1} - y_{i-1}}{2h} = y_i' + O(h^2)\,. \tag{3.19}$$

Thus, the l.h.s. of (3.19) provides a more accurate approximation to y_i' than does the l.h.s. of (3.18). So we use Eq. (3.19) to produce a 2nd-order method:

$$Y_{i+1} = Y_{i-1} + 2hf_i, \tag{3.20}$$

which is of the form (3.17). We need both Y_0 and Y_1 to start this method.

A divergent third-order method
(The term "divergent" will be explained in the next Lecture.)

Let us try to increase the order of method (3.20) from 2nd to 3rd by including extra terms into the scheme:

$$Y_{i+1} - (a_0 Y_i + a_1 Y_{i-1} + a_2 Y_{i-2}) = b_0 h\, f_i\,, \tag{3.21}$$

where we now require that the local truncation error of (3.21) be $O(h^4)$. We can follow the derivation found either in Sec. 3.1 (Taylor-series expansion) or Sec. 3.2 (requiring that (3.21) hold true for $y = p_3(x)$) to obtain the values of the coefficients a_0 through a_2, and b_0. The result is:

$$Y_{i+1} + \frac{3}{2}Y_i - 3Y_{i-1} + \frac{1}{2}Y_{i-2} = 3h\, f_i\,. \tag{3.22}$$

Supposedly, method (3.22) is more accurate than the Leap-frog method (3.20). However, we will show in the next Lecture that method (3.22) is *completely useless* for numerical computations.

3.4 Starting a multistep method

To start any of the single-step methods, considered in Lectures 1 and 2, one only needs to know the initial condition, $Y_0 = y_0$, at $x = x_0$. To start any multistep method, one needs to know the numerical solution at several points. For example, to start an Adams–Bashforth method of order m, one would need the values $Y_0, \ldots, Y_{m-1}$ (see Eqs. (3.5), (3.11), and (3.12)). That is, to start an mth-order method, one needs to know the solution at the first m points. We will now address the following **question**:

[9] You will be asked to verify this.

Suppose that we want to start a multistep method of order m using the values $Y_1, \ldots, Y_{m-1}$ that have been computed by a starting (single-step) method of order n. What should the order n of the starting method be so as **not to compromise the order m of the multistep method**?

First, it is clear that if $n \geq m$, then the local truncation error (LTE) made in the computation of $Y_1, \ldots, Y_{m-1}$ and of the terms on the r.h.s. of (3.16) and (3.17) will be at least as small (in the order of magnitude sense) as the LTE of the multistep method. So, using a starting method whose order is *no less* than the order of the multistep method will not degrade the accuracy of the latter method. But is it possible to use a starting method with $n < m$ for the same end result?

We will now show, using method (3.16) as an example, that it is possible to take $n = m-1$ (i.e., the starting method's order may be one less than the multistep method's order).[10]

The LTEs of Y_1 through Y_{m-1} are $O(h^{n+1})$. Then the error contributed to Y_m from the second term (i.e., from Y_i with $i = m-1$) on the l.h.s. of (3.16), is $O(h^{n+1})$:

$$\text{error of l.h.s. of (3.16)} \;=\; O(h^{n+1}). \tag{3.23}$$

Next, if f_i through f_{i-N} on the r.h.s. were calculated using the exact solution $y(x)$, then the error of the r.h.s. would have been $\underline{O(h^{m+1})}$. Indeed, this error is just the LTE of method (3.16) that arises due to the approximation of $\displaystyle\int_{x_i}^{x_{i+1}} f(x, y(x))dx$ by $\displaystyle h\sum_{k=0}^{N} b_k f_{i-k}$. However, the f_{i-k}'s are calculated using values Y_1 through Y_{m-1} which themselves have been obtained with the error $O(h^{n+1})$ of the starting method. Then the error of each f_{i-k} is also $\underline{O(h^{n+1})}$. [11] Therefore, combining the two underlined errors in the text above, one has:

$$\text{error of r.h.s. of (3.16)} \;=\; O(h^{m+1}) + h\cdot O(h^{n+1}) \;=\; \max\{O(h^{n+2}), O(h^{m+1})\}\,. \tag{3.24}$$

Let us summarize. The error on the l.h.s. of (3.16), which occurs due to Y_i (with $i = m-1$) being computed by an nth-order starting method, is $O(h^{n+1})$. The error on the r.h.s of (3.16) is given by (3.24). Therefore:

$$\text{combined error in (3.16)} \;=\; O(h^{n+1}) \text{ from l.h.s.} + \max\{O(h^{n+2}), O(h^{m+1})\} \text{ from r.h.s.} \;= O(h^{n+1})\,. \tag{3.25}$$

(recall that we are only interested in the situation where $n < m$).

Now, in order not to decrease the accuracy of the multistep method, this error must satisfy two criteria:

(i) It must have the same order of magnitude as the global error at the end of the computation, i.e., $O(h^m)$; and <u>in addition</u>,

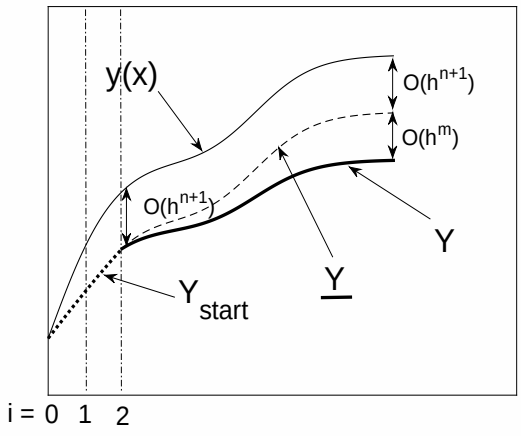
(ii) It may *propagate* to the next computed solution, i.e., to Y_{i+2}, but it *must not accumulate* at each step with other errors of the same magnitude. (See an explanation below.)

[10] Unfortunately, I was unable to find any detailed published proof of this result, and so the derivation found below is my own. As such, it is subject to mistakes ☺. However, a set of Matlab codes accompanying this Lecture where the 3rd-order Adams–Bashforth method (3.11) can be started using the modified Euler, Midpoint, or simple Euler method, shows that if not this derivation itself, than at least its result is probably correct.

[11] This can be seen by the following calculation. Let $j \equiv i-k$ and $Y_j - y(x_j) = O(h^{n+1})$. Then $|f(x_j, Y_j) - f(x_j, y(x_j))| \leq L|Y_j - y(x_j)| = O(h^{n+1})$, where the inequality holds in view of the Lipshitz condition.

One can easily see that criterion (i) is indeed satisfied for $n+1 = m$, i.e., when $n = m-1$. As for criterion (ii), it is also satisfied, although at first sight that might seem strange. Indeed, we have seen in Lecture 1 that if a LTE is $O(h^{n+1})$, then the global error should be $O(1/h) \cdot O(h^{n+1}) = O(h^n)$. However, this is *not so* for the situation considered here. While it is possible to show this by a painstaking analysis of the orders of terms in Eqs. (3.16) and (3.23)–(3.25), it is much easier (although perhaps not as rigorous) to illustrate the situation with a figure, as shown below.

In this figure, $y(x)$ is the exact solution, Y is the numerical solution obtained with a multistep method of order m (where in the figure, $m = 2+1 = 3$), and $Y_{\rm start}$ is the solution obtained by the starting method for the first m steps (recall that Y_0 is the initial condition and thus does not need to be computed by the starting method). The dashed line shows the fictitious solution, denoted as $\underline{Y}$, which starts at $(Y_{\rm start})_m$ and is computed under the hypothesis that the term $\int_{x_i}^{x_{i+1}} f(x, y(x))dx$ is computed exactly rather than being approximated by $h\sum_{k=0}^{N} b_k f_{i-k}$. The magnitude order of the error generated by each of the methods is shown next to the double arrows.



One can see that $\underline{Y}$ is just the exact solution of the ODE except that its initial condition is shifted by $Y_{\rm start}(x_m) - y(x_m) = O(h^{n+1})$. Consequently, it is intuitive to expect that the corresponding final solutions will differ by a similar amount. Thus, $y(x_{\rm final}) - \underline{Y}(x_{\rm final}) = O(h^{n+1})$, and hence at $x = x_{\rm final}$,

$$y(x) - Y = (y(x) - \underline{Y}) + (\underline{Y} - Y) = O(h^{n+1}) + O(h^m).$$

Thus, for $n = m - 1$, the global error will be $O(h^m)$, as required.

Let us summarize. We have shown that in order to have a multistep method (3.16) of order m, the order n of the starting single-step method should be no lower than $(m-1)$. However, in Sec. 3.6 we will see that there is a reason why one may want to use a single-step method of order m (as opposed to $(m-1)$) to start a multistep method of order m.

3.5 Predictor–corrector methods: General form

Let us recall the Modified Euler method introduced in Lecture 1 and write it here using slightly different notations:

$$\begin{aligned} Y^p_{i+1} &= Y_i + hf_i \\ Y^c_{i+1} &= Y_i + \tfrac{1}{2}h\left(f_i + f(x_{i+1}, Y^p_{i+1})\right) \\ Y_{i+1} &= Y^c_{i+1}\,. \end{aligned} \tag{3.26}$$

We can interpret the above as follows: We first *predict* the new value of the solution Y_{i+1} by the first equation, and then *correct* it by the second equation. Methods of this kind are called *predictor–corrector* (P–C) methods.

Question: What is the optimal relation between the orders of the predictor and corrector equations?

Answer: The example of the Modified Euler method suggests that the order of the corrector should be one higher than that of the predictor. More precisely, the following theorem holds:

Theorem 3.1 If the order of the corrector equation is n, then the order of the corresponding P–C method is also n, provided that the order of the predictor equation is no less than $n-1$.

Proof We will assume that the global error of the corrector equation by itself is $O(h^n)$ and the global error of the predictor equation by itself is $O(h^{n-1})$. Then we will prove that the global error of the combined P–C method is $O(h^n)$.

The general forms of the predictor and corrector equations are, respectively:

$$\text{Predictor:} \qquad Y^p_{i+1} = Y_{i-Q} + h\sum_{k=0}^{N} p_k f_{i-k}, \tag{3.27}$$

$$\text{Corrector:} \quad Y^c_{i+1} = Y_{i-D} + h\sum_{k=0}^{M} c_k f_{i-k} + hc_{-1} f(x_{i+1}, Y^p_{i+1})\,. \tag{3.28}$$

In the above two equations, Q, D, N, M are some integer nonnegative numbers. (One of the questions at the end of this Lecture asks you to represent Eq. (3.26) in the form (3.27), (3.28), i.e. to give values for Q, D, N, M and the coefficients p_k's and c_k's.)

You may recognize the predictor equation (3.27) as being an Adams–Bashforth method (3.16). The corrector equation represents another family of multistep methods called Adams–Moulton. Their more precise formulation will be given in Sec. 3.7.

Let us begin by summarizing what we know about the LTE of the predictor equation (3.27). As we have done in previous derivations, let us assume that all computed values Y_{i-k}, $k = 0, 1, 2, \ldots$ coincide with the exact solution at the coresponding points: $Y_{i-k} = y_{i-k}$. Then we can use the identity

$$y_{i+1} = y_{i-Q} + (y_{i+1} - y_{i-Q}) \overset{\text{see (3.13)}}{=} y_{i-Q} + \int_{x_{i-Q}}^{x_{i+1}} y'(x)dx = Y_{i-Q} + \int_{x_{i-Q}}^{x_{i+1}} f(x, y(x))dx$$

and rewrite Eq. (3.27) as:

$$Y^p_{i+1} = Y_{i-Q} + \int_{x_{i-Q}}^{x_{i+1}} f(x, y(x))dx + \left(h\sum_{k=0}^{N} p_k f_{i-k} - \int_{x_{i-Q}}^{x_{i+1}} f(x, y(x))dx \right) \quad \Rightarrow \quad Y^p_{i+1} = y_{i+1} + E_P\,. \tag{3.29}$$

Here E_P is the error made by replacing the exact integral

$$\int_{x_{i-Q}}^{x_{i+1}} f(x, y(x))dx$$

by the linear combination of f_{i-k}'s, found on the r.h.s. of (3.27). Since, by the condition of the Theorem, the global error of the predictor equation is $O(h^{n-1})$, then the LTE E_P has the order of $O(h^{(n-1)+1}) = O(h^n)$.

We are now ready to establish the LTE of the corrector equation (3.28) following similar lines. Namely, similarly to (3.29), Eq. (3.28) can be rewritten as

$$Y^c_{i+1} = y_{i+1} + E_C + hc_{-1}\left(f(x_{i+1}, Y^p_{i+1}) - f(x_{i+1}, y_{i+1}) \right)\,. \tag{3.30}$$

Here E_C is the error obtained by replacing the exact integral

$$\int_{x_{i-D}}^{x_{i+1}} f(x, y(x))dx$$

by the quadrature formula

$$h \sum_{k=-1}^{M} c_k f_{i-k}$$

(note that the lower limit of the summation is different from that in (3.28)!). The last term on the r.h.s. of (3.30) occurs because, unlike all previously computed Y_{i-k}'s, the $Y^p_{i+1} \neq y_{i+1}$.

To complete the proof,[12] we need to show that $Y^c_{i+1} - y_{i+1} = O(h^{n+1})$ in (3.30). By the condition of the Theorem, the corrector equation has order n, and hence the LTE $E_C = O(h^{n+1})$. Then all that remains to be estimated is the last term on the r.h.s. of (3.30). To that end, we recall that f satisfies the Lipschitz condition with respect to y, whence

$$|f(x_{i+1}, Y^p_{i+1}) - f(x_{i+1}, y_{i+1})| \le L|Y^p_{i+1} - y_{i+1}| = L\,|E_P|, \tag{3.31}$$

where L is the Lipschitz constant. Combining Eqs. (3.30) and (3.31) and using the triangle inequality (1.5), we finally obtain

$$|Y^c_{i+1} - y_{i+1}| \le |E_C| + hL|E_P| = O(h^{n+1}) + h \cdot O(h^n) = O(h^{n+1})\,, \tag{3.32}$$

which proves that the P–C method has the LTE of order $n+1$, and hence is the nth-order method. ***q.e.d.***

We now present two P–C pairs that in applications are sometimes preferred[13] over the Modified Euler method. The first pair is:

$$\begin{aligned} &\text{Predictor:} && \text{2nd-order Adams–Bashforth} \\ &Y^p_{i+1} = && Y_i + \tfrac{1}{2}h(3f_i - f_{i-1}) \\ &\text{Corrector:} && \text{2nd-order Adams–Moulton (same as Trapezoidal rule)} \\ &Y^c_{i+1} = && Y_i + \tfrac{1}{2}h\left(f_i + f^p_{i+1}\right)\,, \end{aligned} \tag{3.33}$$

where $f^p_{i+1} = f(x_{i+1}, Y^p_{i+1})$. The order of the P–C method (3.33) is two.

The other pair, of the 4th-order accuracy, is:

$$\begin{aligned} &\text{Predictor:} && \text{4th-order Adams–Bashforth} \\ &Y^p_{i+1} = && Y_i + \tfrac{1}{24}h(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3})\,. \\ &\text{Corrector:} && \text{4th-order Adams–Moulton} \\ &Y^c_{i+1} = && Y_i + \tfrac{1}{24}h(9f^p_{i+1} + 19f_i - 5f_{i-1} + f_{i-2})\,, \end{aligned} \tag{3.34}$$

In all P–C methods it is implied that $Y_{i+1} \equiv Y^c_{i+1}$.

3.6 Predictor–corrector methods: Error monitoring

An observation one can make from Eqs. (3.33) is that *both* the predictor and corrector equations have the order two (i.e. the LTEs of $O(h^3)$). In view of the Theorem of the previous subsection, this may seem to be unnecessary. Indeed, the contribution of the predictor's error to the LTE is $h \cdot O(h^3) = O(h^4)$ (see Eq. (3.32)), while the LTE of the corrector equation

[12] At this point, you have probably forgotten what we are proving. Pause, re-read the Theorem's statement, and then come back to finish the reading.

[13] In the next Section we will explain why this is so.

itself (which determines that of the entire P–C method) is only $O(h^3)$. There is, however, an important consideration because of which method (3.33) may be preferred over the Modified Euler. Namely, one can monitor the error size in (3.33), whereas the Modified Euler does not give its user such a capability. Below we explain this statement in detail. A similar treatment can be applied to the 4th-order method (3.34).

The key fact is that the LTEs of the predictor and correction equations (3.33) are *proportional to each other* in the leading order:

$$y_{i+1} - Y^p_{i+1} = \quad \tfrac{5}{12}h^3 y'''_i + O(h^4), \tag{3.35}$$

$$y_{i+1} - Y^c_{i+1} = \quad -\tfrac{1}{12}h^3 y'''_i + O(h^4)\,. \tag{3.36}$$

For the reader's information, the analogues of the above estimates for the 4th-order method (3.34) are:

4th-order P–C method (3.34):

$$y_{i+1} - Y^p_{i+1} = \qquad \frac{251}{720}h^5 y^{(5)}_i + O(h^6),$$

$$y_{i+1} - Y^c_{i+1} = \qquad -\frac{19}{720}h^5 y^{(5)}_i + O(h^6)\,.$$

We derive (3.36) in Appendix 1 to this Lecture, while the derivation of (3.35) is left as an exercise. Here we only note that the **derivation of (3.36) hinges upon the fact that $y_i - Y^p_i = O(h^3)$, which is guaranteed by (3.35). Otherwise, i.e. if $y_i - Y^p_i = O(h^2)$, as in the predictor for the Modified Euler method, the term on the r.h.s. of (3.36) would not have had such a simple form.** See Remarks 2 and 3 after Eq. (3.40) for details.

We will now explain how (3.35) and (3.36) can be used together to monitor the error of the P–C method (3.33). From (3.36) we obtain the error of the corrector equation:

$$|\epsilon^c_{i+1}| \approx \frac{1}{12}h^3 |y'''_i|\,. \tag{3.37}$$

On the other hand, from Eqs. (3.35) and (3.36) together, we have

$$|Y^p_{i+1} - Y^c_{i+1}| \approx \left(\frac{5}{12} + \frac{1}{12}\right) h^3 |y'''_i|. \tag{3.38}$$

Thus, from (3.37) and (3.38) one can estimate the error via the difference of the predicted and corrected values of the solution:

$$|\epsilon^c_{i+1}| \approx \frac{1}{6}|Y^p_{i+1} - Y^c_{i+1}|\,. \tag{3.39}$$

Moreover, Eqs. (3.35) and (3.36) can also be used to obtain a higher-order method than (3.33), because they imply that

$$y_{i+1} = \frac{1}{6}\left(Y^p_{i+1} + 5Y^c_{i+1}\right) + O(h^4)\,.$$

Hence

$$Y_{i+1} = \frac{1}{6}\left(Y^p_{i+1} + 5Y^c_{i+1}\right) \tag{3.40}$$

produces a more accurate approximation to the solution than either Y^p_{i+1} or Y^c_{i+1} alone. (Note a similarity with the Romberg extrapolation described in Lecture 1.)

Thus, Eqs. (3.33), (3.39), and (3.40) can be used to program a P–C method that allows the user to monitor the error. Namely:

1. Compute Y_i^p and Y_i^c from (3.33). Compute the improved solution from (3.40).
2. Estimate the error of Y_{i+1}^c using (3.39).

The above procedure produces a 3rd-order-accurate solution (3.40) while monitoring the error size of the associated 2nd-order method (3.33). This is analogous to the situation for the adaptive RK methods described in Lecture 2, which computed an nth-order accurate solution while controlling the error of a related method of the lower order, $(n-1)$.

The following Remarks address various issues that arise when at least one of the methods in the P–C pair is multistep.

Remark 1: As we have just discussed, using the predictor and corrector equations of the same order has the advantage of allowing one to monitor the error. However, it may have a disadvantage of making such schemes less *stable* compared to schemes where the predictor's order is one less than that of the corrector. We will study the concept of stability in the next Lecture.

Remark 2: Suppose we plan to use a P–C method with the predictor and corrector equations having the same order, say m, so as to monitor the error, as described above. Furthermore, suppose that the predictor and/or corrector equations are based on an Adams–Bashforth multistep method, i.e. that of the form (3.16). (For example, the predictor equations in (3.33) and (3.34) are of this form.) Let us now **re-examine the question** addressed in Sec. 3.4, namely: **What order starting method should we use in the predictor and corrector equations *so that the P–C method would be able to monitor the error?***

Let us begin with the predictor, which is one of the Adams–Bashforth methods. In Sec. 3.4 we showed that a starting method of order $(m-1)$ would suffice to make an Adams–Bashforth method have order m. Suppose now that we have a P–C method (3.27), (3.28) based on a pair of Adams–Bashforth methods (i.e., with $Q = D = 0$), like (3.33) or (3.34), and we want to be able to monitor its error (as in (3.39)) and construct a solution more accurate than both the predictor and corrector equations (as in (3.40)). Below we examine the question: What order of the starting method should we use to start the predictor and corrector equations?

Let us do so for the specific example of the P–C method (3.33). Suppose we start the predictor method by the simple Euler. Then Y_0 (the initial condition) is exact, and the LTE of Y_1 is $O(h^2)$. Starting with $i = 1$, the predictor equation computes Y_{i+1}^p with LTE of the same order, i.e., $O(h^2)$: As explained in Section 3.4, this error will propagate to Y_i^p for any i, but will keeps its order (i.e., will not become $O(h)$). While this will not affect the accuracy of the predictor equation, it will destroy the form of the $O(h^3)$-error in (3.35), and hence will invalidate (3.39) and (3.40).

If, instead, we start the (2nd-order) predictor equation in (3.33) with a 2nd-order starting method, the LTE of Y_1, which will propagate to all subsequent Y_i^p, will be $O(h^3)$ instead of $O(h^2)$. This will still destroy the form of the $O(h^3)$-LTE in (3.35). Hence, in order to be able to monitor the error and have a higher-order solution, as in (3.40), the starting order of the 2nd-order predictor equation must be 3rd-order! The same considerations apply to a starting method for the corrector equation. Then, the error in finding Y_1 will be $O(h^4)$, and the form of the $O(h^3)$-terms in both (3.35) and (3.36) will be preserved.

Thus, generalizing the above, we conclude: **If you want to be able to monitor the error in a P–C method where the predictor and corrector equations are, respectively, Adams–Bashforth- and Adams–Moulton-type multistep methods of order m, you need to start both of these methods by a singlestep method of**

order $(m+1)$ or higher.[14]

Similar considerations show that this conclusion will remain the same if the equations in the P–C method have the general form (3.27), (3.28) with $Q > 0$ and/or $D > 0$.

Remark 3: At the end of Sec. 3.1 we listed disadvantages of using Adams–Bashforth or (3.17)-type multistep methods with a variable step size. Let us reiterate here one of them. Namely, the r.h.s.'s of (3.35) and (3.36) are not simply proportional to y''' but contain other terms if the step size is varied. Therefore, even error monitoring for P–C methods with a variable step size is a complicated task.

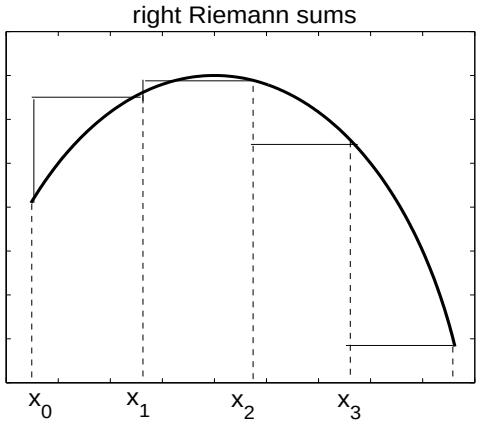
To summarize on the P–C methods:

1. The P–C methods may provide both high accuracy and the capability of error monitoring, all at a potentially lower computational cost than the RK–Fehlberg method (RKF). For example, the P–C method (3.34) has the error of the same (fourth) order as the RKF, while requiring $k+1$ function evaluations, where k is the number of times one has to iterate the corrector equation. If $k < 4$, then the 4th-order P–C method (3.34) requires fewer function evaluations than the RKF.

2. The adjustment of the step size in P–C methods is a complicated task (as it is in all multistep methods); see the end of Sec. 3.1 and Remark 3 above. If one really needs to use P–C methods with a variable step, the reader is advised to first read Appendix 3, where the idea of, essentially, a reformulation of Adams–Moulton methods in a manner that allows step size varying, is explained. One can then proceed to learning the documentation of the LSODE and VODE methods, mentioned there and based on the same idea, and use these professionally developed packages.

3.7 Implicit methods

We noted in Lecture 1 that the simple Euler method is analogous to the left Riemann sums when integrating the differential equation $y' = f(x)$.



The method analogous to the right Riemann sums is:

$$Y_{i+1} = Y_i + hf(x_{i+1}, Y_{i+1})\,. \qquad (3.41)$$

It is called the implicit Euler, or backward Euler, method. This is a first-order method: Its global error is $O(h)$ and the LTE is $O(h^2)$.

We note that if $f(x,y) = a(x)y + b(x)$, then the implicit equation (3.41) can be easily solved:

$$Y_{i+1} = \frac{Y_i + hb_{i+1}}{1 - ha_{i+1}}\,. \qquad (3.42)$$

However, for a general nonlinear $f(x,y)$, equation (3.41) cannot be solved exactly, and its solution then has to be found numerically, say, by the Newton–Raphson method.

[14] The above consideration was inspired by an observation made by Mr. Tyler Gray, who took this course in Spring 2016.

Question: Why does one want to use the implicit Euler, which is so much harder to solve than the simple Euler method?

Answer: Implicit methods have *stability* properties that are much better than those of explicit methods (like the simple Euler). We will discuss this in the next Lecture.

This increased stability had provided a motivation for researches to construct implicit multistep methods. One of the most well-known examples of such methods is the Adams–Moulton family, mentioned in Sec. 3.5. Its general form is very similar to that of Adams–Bashforth methods, Eq. (3.16):

$$Y_{i+1} - Y_i = h \sum_{k=-1}^{N} b_k f_{i-k}. \tag{3.43}$$

Can you see why these methods are implicit?

In Lecture 4 you will learn that adding an Adams–Moulton Corrector to an Adams–Bashforth Predictor, as in (3.34), makes this P–C combination more stable than the Adams–Bashforth alone.

Finally, we present equations for the Modified implicit Euler method:

$$Y_{i+1} = Y_i + \frac{h}{2}\left(f(x_i, Y_i) + f(x_{i+1}, Y_{i+1})\right) . \tag{3.44}$$

This is a second-order method.

3.8 Appendix 1: Derivation of (3.36)

Here we derive the LTE of the corrector equation in the method (3.33). Assuming, as usual, that $Y_i = y_i$, and using $Y^p_{i+1} = y_{i+1} + O(h^3)$ (since the order of the predictor equation is two), one obtains from the corrector equation of (3.33):

$$\begin{aligned}
Y^c_{i+1} = &\quad y_i + \tfrac{1}{2}h\left(y'_i + f(x_{i+1}, y_{i+1} + O(h^3))\right) \\
= &\quad y_i + \tfrac{1}{2}h\left(y'_i + f(x_{i+1}, y_{i+1}) + O(h^3)\right) \\
= &\quad y_i + \tfrac{1}{2}h\left(y'_i + y'_{i+1} + O(h^3)\right) \\
= &\quad y_i + \tfrac{1}{2}h\left(y'_i + \left[y'_i + hy''_i + \tfrac{1}{2}h^2 y'''_i + O(h^3)\right] + O(h^3)\right) \\
= &\quad y_i + hy'_i + \tfrac{1}{2}h^2 y''_i + \tfrac{1}{4}h^3 y'''_i + O(h^4) .
\end{aligned} \tag{3.45}$$

On the other hand, for the exact solution we have the usual Taylor series expansion:

$$y_{i+1} = y_i + hy'_i + \frac{1}{2}h^2 y''_i + \frac{1}{6}h^3 y'''_i + O(h^4) . \tag{3.46}$$

Subtracting (3.45) from (3.46), we obtain

$$y_{i+1} - Y^c_{i+1} = -\frac{1}{12}h^3 y'''_i + O(h^4) ,$$

which is (3.36).

3.9 Appendix 2: Matrix form of multistep methods

The main reason *why* one wants to recast the rather clear form (3.17) into something else (which may even not look as clear at first) will be explained in the next Lecture. This other form will also be used in Appendix 3. Here we will merely illustrate *how* the matrix form of a multistep method can be obtained, using as an example a particular case of (3.17):

$$Y_{i+1} - a_0 Y_i - a_1 Y_{i-1} = h\left[b_0 f_i + b_1 f_{i-1} + b_2 f_{i-2} + b_3 f_{i-3}\right]; \tag{3.47}$$

thus, in the notations of (3.17), $M = 1$ and $N = 3$. Namely, you should verify that (3.47) is equivalent to:

$$\vec{\mathbf{y}}_{i+1} = \begin{pmatrix} a_0 & a_1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \vec{\mathbf{y}}_i + h \begin{pmatrix} b_0 & b_1 & b_2 & b_3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} f_i \\ f_{i-1} \\ f_{i-2} \\ f_{i-3} \end{pmatrix}, \qquad \text{where } \vec{\mathbf{y}}_i \equiv \begin{pmatrix} Y_i \\ Y_{i-1} \\ Y_{i-2} \\ Y_{i-3} \end{pmatrix}. \tag{3.48}$$

Note that the dimension of $\vec{\mathbf{y}}_i$ equals $\max(M+1, N+1)$. For example, if in (3.47) one has $b_1 = b_2 = b_3 = 0$, so that $N = 0$, the corresponding version of (3.48) is:

$$\vec{\mathbf{y}}_{i+1} = \begin{pmatrix} a_0 & a_1 \\ 1 & 0 \end{pmatrix} \vec{\mathbf{y}}_i + h \begin{pmatrix} b_0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} f_i \\ f_{i-1} \end{pmatrix}, \qquad \text{where now } \vec{\mathbf{y}}_i \equiv \begin{pmatrix} Y_i \\ Y_{i-1} \end{pmatrix}. \tag{3.49}$$

Note that putting a multistep method into a matrix form as above achieves the following: It makes this method *singlestep* for a vector quantity $\vec{\mathbf{y}}_i$. This fact will be used in Appendix 3.

3.10 Appendix 3: The idea behind the Nordsieck method for multistep methods with adaptive step size

This presentation is based on the paper [A. Nordsieck, "On numerical integration of ordinary differential equations," Math. Comp., v. 16, p. 22 (1962)], posted next to this Lecture. Its main point is: How can one reformulate a multistep method to avoid the difficulty of varying the time step mentioned at the end of Sec. 3.1. For practical reasons, Nordsieck focuses on the case of a 5-stage multistep method. However, for the sake of the clarity of the exposition, I'll work out the case of the 3-stage method. The IVP that we will solve is, as before,

$$y'(x) = f(x, y), \qquad y(x_0) = y_0\,. \tag{1.1}$$

It should be noted that a suite of codes based on Nordsieck's method was developed by professional software engineers at the Lawrence Livermore National Laboratory in the 1990s. Here is a a link to their depository at the LLNL website (different from the link given in Sec. 3.1): `https://computing.llnl.gov/projects/odepack`. The link to the software is near the top-right corner of the page. I am not aware of a comprehensive performance comparison between the ODEPACK (the above software package) with Matlab's suite of ODE solvers, whose description is found here: `https://www.mathworks.com/help/matlab/math/choose-an-ode-solver.html` (under "Basic Solver Selection").

The motivation for Nordsieck's method is to avoid the difficulties that arise when the step size is varied in a multistep method. Then one may think that those difficulties will be avoided if a multistep method is recast in the form of a singlestep one. Indeed, we saw in Sec. 2.2 that

it is (much more) straightforward to adjust the step size in a singlestep RK method. We have also seen in Appendix 3 how a multistep method can be recast as a singlestep one. Nordsieck's idea, however, was different in some technical details from that used in Appendix 3. Namely, instead of creating a vector of the solution at several consecutive steps, he proposed to create a vector of quantities that closely approximate the *derivatives* of the solution at a given point, x_i. (The zeroth derivative is, of course, the solution $y(x_i)$ itself.) As we saw in Sec. 3.1, those derivatives are directly related to certain combinations of $f_{i-k} \equiv f\big(x_{i-k}, y(x_{i-k})\big)$.

We will begin by seeking a numerical solution of (1.1) whose local truncation error (LTE) is $O(h^5)$; so that our method is expected to be of the 4th order. The exact solution has the Taylor expansion

$$y_{i+1} = y_i + h\, f_i + \frac{h^2}{2!}\, y_i'' + \frac{h^3}{3!}\, y_i''' + \frac{h^4}{4!}\, y_{i,*}^{(4)}, \tag{3.50}$$

where $y_{i,*}^{(4)}$ is the 4th derivative of the solution (w.r.t. x) at some point $x_{i,*} \in [x_i, x_{i+1}]$. Denoting (slightly differently than Nordsieck did)

$$a_i = \frac{h^2}{2!}\, y_i'', \qquad b_i = \frac{h^3}{3!}\, y_i''', \qquad c_i = \frac{h^4}{4!}\, y_{i,*}^{(4)}, \tag{3.51}$$

we can write the *first equation* of our new method as:

$$Y_{i+1} = Y_i + h\, f_i + a_i + b_i + c_i. \tag{3.52}$$

If we now obtain 3 more equation for f_{i+1}, a_{i+1}, and b_{i+1} and somehow eliminate c_i, we will have an equation for one step for the vector $[Y,\, f,\, a,\, b]^T$ (where the superscript denotes the transposition). This is how we will proceed in the next paragraph but will end up with a slightly different result.

The equation for f_{i+1} follows from differentiating (3.50) once and using the ODE (1.1) and the notations (3.50):

$$h\, f_{i+1} = h\, f_i + 2a_i + 3b_i + 4\, \frac{h^4}{4!}\, y_{i,**}^{(4)}. \tag{3.53}$$

Note that the last term is not quite equal to $4c_i$ because the derivative is evaluated at some point $x_{i,**} \in [x_i, x_{i+1}]$ and, in general, $x_{i,**} \neq x_{i,*}$ because they pertain to Taylor expansions of different functions (f and y, respectively). Nonetheless, given that $x_{i,**} = x_{i,*} + O(h)$ (and assuming sufficient smoothness of the solution $y(x)$), one has:

$$\frac{h^4}{4!}\, y_{i,**}^{(4)} = c_i + O(h^5). \tag{3.54}$$

A *key step* in Nordsieck's construction of the method is to neglect the $O(h^5)$ term above and hence obtain the expression for c_i from (3.53):

$$c_i = \frac{h}{4}\, [f_{i+1} - f^p]\,, \qquad \text{where} \quad h\, f^p \equiv h\, f_i + 2a_i + 3b_i \tag{3.55}$$

is a so-called predicted value (multiplied by h) for f_{i+1}. Then, similarly to (3.53) and (3.54), one obtains:

$$a_{i+1} = a_i + 3b_i + 6c_i + O(h^5), \qquad b_{i+1} = b_i + 4c_i + O(h^5). \tag{3.56}$$

Finally, ignoring the $O(h^5)$ terms in (3.56) and combining Eqs. (3.52), (3.55), and (3.56), one has the following *tentative* form of Nordsieck's 3-stage method:

$$Y_{i+1} = Y_i + h\, f_i + a_i + b_i + \frac{h}{4}\, (f_{i+1} - f^p)\,, \tag{3.57a}$$

$$a_{i+1} = a_i + 3b_i + \frac{6h}{4}\,(f_{i+1} - f^p)\,, \tag{3.57b}$$

$$b_{i+1} = b_i + \frac{4h}{4}\,(f_{i+1} - f^p)\,, \tag{3.57c}$$

with f^p being given by (3.55).

We have called method (3.57) 'tentative' because with the coefficients in front of the terms in parentheses as there, it can be shown (using the material learned in Lecture 4) to be completely useless for numerical computation. The reason for its uselessness is that this method is *unstable*, just as the 'Divergent third-order method' at the end of Sec. 3.3 is; the concept of (in)stability will be discussed Lecture 4. To render the method stable (and hence useful), one needs to change the coefficients $\kappa_y = 1/4$, $\kappa_a = 6/4$, $\kappa_b = 4/4$ in (3.57) to $\kappa_y = 3/8$, $\kappa_a = 3/4$, $\kappa_b = 1/6$, respectively. We will justify this choice in Appendix 3 of Lecture 4. For now, we will present the *actual* form of Nordsieck's 3-stage method:

$$Y_{i+1} = Y_i + h\,f_i + a_i + b_i + \frac{3h}{8}\,(f_{i+1} - f^p)\,, \tag{3.58a}$$

$$a_{i+1} = a_i + 3b_i + \frac{3h}{4}\,(f_{i+1} - f^p)\,, \tag{3.58b}$$

$$b_{i+1} = b_i + \frac{h}{6}\,(f_{i+1} - f^p)\,. \tag{3.58c}$$

Note that this method is implicit, since $f_{i+1} \equiv f(x_{i+1}, Y_{i+1})$.

We will now address the (highly nontrivial!) issue of the accuracy of method (3.58). At first sight, one can surmise that its LTE is only $O(h^4)$. Indeed, the truncation error in (3.52), from which the "primordial" method (3.57) was obtained, is $O(h^5)$. However, by "distorting" the coefficients $\kappa_{y,a,b}$, as stated above, one should in general expect to lose one order of accuracy, i.e. increase the LTE from $O(h^5)$ to $O(h^4)$. However, by careful design due to Nordsieck, whose idea will be explained in Appendix 3 of Lecture 4, this turns out not to be the case, and the LTE of Nordsieck's 3-stage method (3.58) can be shown to be $O(h^5)$! We will *not* establish such a nontrivial fact, requiring heavy calculations, in this conceptual account of Nordsieck's method but will instead give a couple of reasons why this is indeed so.

As one plausible reason, note that while a_i, b_i in the primordial method (3.57) are exactly proportional to the derivatives of the solution (see (3.51)), in the actual method (3.58) they are only approximately (i.e., in the main order) proportional to those derivatives; this is because the coefficients of the last terms in (3.58) have been "distorted." See Eqs. (14) in Nordsieck's paper for the precise meaning of the aforementioned approximation. Then, those smaller, next-order-in-h corrections magically combine so as to nullify the $O(h^4)$ terms in the LTE and thus yield a $O(h^5)$ LTE.

Another, and explicit, way to demonstrate that the LTE for (3.58) is $O(h^5)$ is to show that this method is equivalent to another method whose LTE is already known to be of that order. This "other method" turns out to be an Adams–Moulton method of the respective order. To demonstrate this, it will be instructive to start with the 2-stage Nordsieck method and then move on to the 3-stage one; this will make it clear what kind of "magic" occurs in connecting Nordsieck's methods to those of the Adams–Moulton family.

The 2-stage Nordsieck method is:

$$Y_{i+1} = Y_i + h\,f_i + a_i + \frac{5h}{12}\,(f_{i+1} - f^p)\,, \tag{3.59a}$$

$$a_{i+1} = a_i + \frac{h}{2}\,(f_{i+1} - f^p)\,, \tag{3.59b}$$

where now the "predicted" value f^p is of a lower (by one) order of accuracy, satisfying:

$$h\,f^p = h\,f_i + 2a_i. \tag{3.59c}$$

Substituting (3.59c) into (3.59b) one finds:

$$a_{i+1} = \frac{h}{2}\,(f_{i+1} - f_i)\,; \tag{3.60}$$

substituting this and, again, (3.59c) into (3.59a) yields:

$$Y_{i+1} = Y_i + \frac{h}{12}\,(5f_{i+1} + 8f_i - f_{i-1})\,. \tag{3.61}$$

This is the 3rd-order Adams–Moulton method (not previously mentioned in these Notes); its LTE is known to be $O(h^4)$. Note that the magic that has allowed one to eliminate a_i from the equation for Y_{i+1} is the fact that a_{i+1} can be expressed *only* in terms of f, but *not* of a_i; see (3.60).

Let us now turn to the 3-stage method (3.58). Substituting $h\,f^p$ from (3.55) into that method one obtains:

$$Y_{i+1} = Y_i + h\,f_i + \frac{1}{4}\left(a_i - \frac{b_i}{2}\right) + \frac{3h}{8}\,(f_{i+1} - f_i)\,, \tag{3.62a}$$

$$\begin{pmatrix} a \\ b \end{pmatrix}_{i+1} = \begin{pmatrix} -1/2 & 3/4 \\ -1/3 & 1/2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix}_i + h\,(f_{i+1} - f_i) \begin{pmatrix} 3/4 \\ 1/6 \end{pmatrix}. \tag{3.62b}$$

It may seem that the magic that worked for the 2-stage method has disappeared for the 3-stage one, because now $[a,\ b]^T_{i+1}$ depends on $[a,\ b]^T_i$. However, note that in (3.62a) one has a *certain combination* of a_i and b_i. To obtain an equation for that combination, multiply (3.62b) on the left by the row $[1,\ -1/2]$ to get:

$$[1,\ -1/2] \begin{pmatrix} a \\ b \end{pmatrix}_{i+1} = [-1/3,\ 1/2] \begin{pmatrix} a \\ b \end{pmatrix}_i + \frac{2h}{3}\,(f_{i+1} - f_i)\,. \tag{3.63}$$

Our difficulty is not yet resolved because the l.h.s. in (3.63) still depends on a combination of a_i, b_i on the r.h.s. We then obtain an equation for this second combination by multiplying (3.62b) on the left by the row $[-1/3,\ 1/2]$, and the result is:

$$[-1/3,\ 1/2] \begin{pmatrix} a \\ b \end{pmatrix}_{i+1} = -\frac{h}{6}\,(f_{i+1} - f_i)\,. \tag{3.64}$$

The magic has reoccurred! That is, by substituting (3.64) into (3.63) and the result into (3.62a), one finally eliminates a_i, b_i and obtains:

$$Y_{i+1} = Y_i + \frac{h}{24}\,(9f_{i+1} + 19f_i - 5f_{i-1} + f_{i-2})\,. \tag{3.65}$$

This is the 4th-order Adams–Moulton method (see (3.34)), whose LTE is $O(h^5)$.

3.11 Questions for self-assessment

1. Make sure you can reproduce the derivation of Eq. (3.4).
2. What is the idea behind the derivation of Eq. (3.5)?
3. Derive Eqs. (3.9) and (3.10).
4. Derive Eq. (3.11) as indicated in the text.
5. Describe the *ideas behind* two alternative ways, as found in Secs. 3.1 and 3.2, to derive formulae for multistep methods.
6. Verify Eq. (3.19).
7. For a multistep method of order m, what should the order of the starting method be?
8. Convince yourself that method (3.26) is of the form (3.27) and (3.28).
9. What is the origin of the error E_P in Eq. (3.29)?
10. What is the origin of the error E_C in Eq. (3.30)?
11. How should the orders of the predictor and corrector equations be related? Why?
12. Is there a reason to use a predictor as accurate as the corrector?
13. What are the advantages and disadvantages of the P–C methods compared to the RK methods?
14. What is the reason one may want to use an implicit method?
15. Verify the second and third equations in Appendix 2 starting from its first equation.
16. Why is the "primordial" form (3.57) of Nordsieck's method useless?
17. Verify equations (3.60) and (3.61).