Adaptive explicit RK-method.

Given an embedded RK-pair:

$$
\begin{array}{c|cccc}
c_2 & a_{21} & & & \\
\vdots & \vdots & & & \\
c_s & a_{s1} & \cdots & a_{s,s-1} & \\
\hline
 & b_1 & \cdots & b_{s-1} & b_s \\
\hline
 & \hat{b}_1 & \cdots & \hat{b}_{s-1} & \hat{b}_s
\end{array}
\quad
\begin{array}{l}
\\ \\ \\
\text{Order } p \\
\text{Order } p+1
\end{array}
$$

$(b_s$ or $\hat{b}_{s-1} = 0)$

Algorithm

Input: f, t_0, t_{end}, y_0, Tol, h_0

$t = t_0$, $y = y_0$, $h = h_0$

while $t < t_{end}$

$$k_i = f(t + c_i h, y + h\sum_{j=1}^{s} a_{ij} k_j), \quad i = 1, \cdots, s$$

$$le_{n+1} = h\sum_{i=1}^{s} (\hat{b}_i - b_i) k_i \qquad \text{(Local error)}$$

if $\|le_{n+1}\| \leq Tol$ (accept the step)

$$y \leftarrow y + h\sum_{i=1}^{s} b_i k_i$$

(or $\hat{b}_i$ if local extrapolation is used)

$$t \leftarrow t + h$$

end

$$h \leftarrow P \cdot \sqrt[p+1]{\frac{Tol}{\|le_{n+1}\|}} \cdot h$$

end

See Wikipedia: List of Runge-Kutta methods for examples.