Résolution d’équations différentielles ordinaires

A la fin du chapitre, l’étudiant doit être capable de:

1. Mettre en œuvre les schémas d’intégration d’Euler, Crank-Nicolson, Adams-Bashforth et Runge-Kutta pour les équations différentielles ordinaires du premier ordre
2. Calculer l’ordre d’une méthode de résolution d’une équation différentielle ordinaire du premier ordre
3. Mesurer numériquement l’ordre d’une méthode de résolution d’une équation différentielle ordinaire du premier ordre

Problème type abordé dans ce chapitre

- Dans de nombreux cas, un effort de modélisation ou une démarche de type ingénieur se solde par un problème aux valeurs initiales du type:

La fonction $\boldsymbol{F}$ étant donnée, trouver la fonction inconnue $\boldsymbol{y(t)}$ telle que $\boldsymbol{\frac{dy}{dt} = F(t,y)}$ pour $\boldsymbol{t > t^0}$, avec la cdt initiale $\boldsymbol{y(t^0) = y^0}$

- Exemples ($t^0 = 0$):
  - Principe fondamental de la dynamique:

$$\frac{dV}{dt} = Force/masse, \qquad V(0) = V_0$$

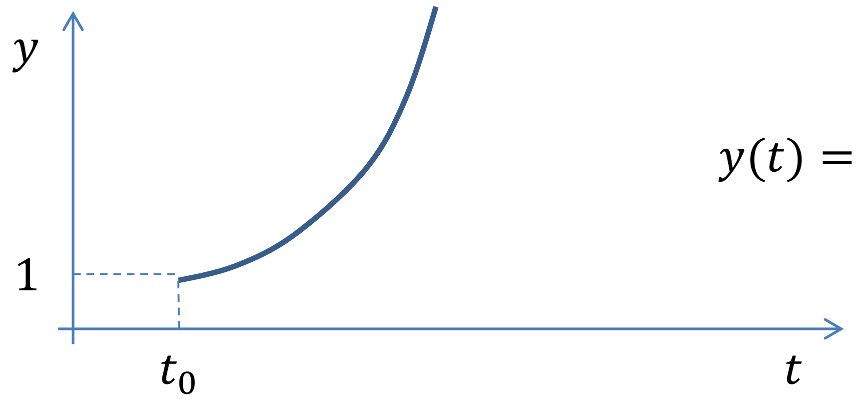
  - Processus chimique

$$\frac{dC}{dt} = -aC^b, \qquad C(0) = C_0$$

Problème de résolution

- Dans certains cas simples, il est possible d'obtenir LA solution analytique de l'équation:

ex: $\frac{dy}{dt} = yt, \quad \text{pour } t > t^0, \text{avec la cdt initiale } y(t^0) = 1,$



$$y(t) = \exp\frac{t^2 - {t^0}^2}{2}$$

- Mais ce n'est en général pas le cas

ex: $\frac{dy}{dt} = \sqrt{y + t}, \quad \text{pour } t > t^0, \text{avec la cdt initiale } y(t^0) = 1$

Discrétisation

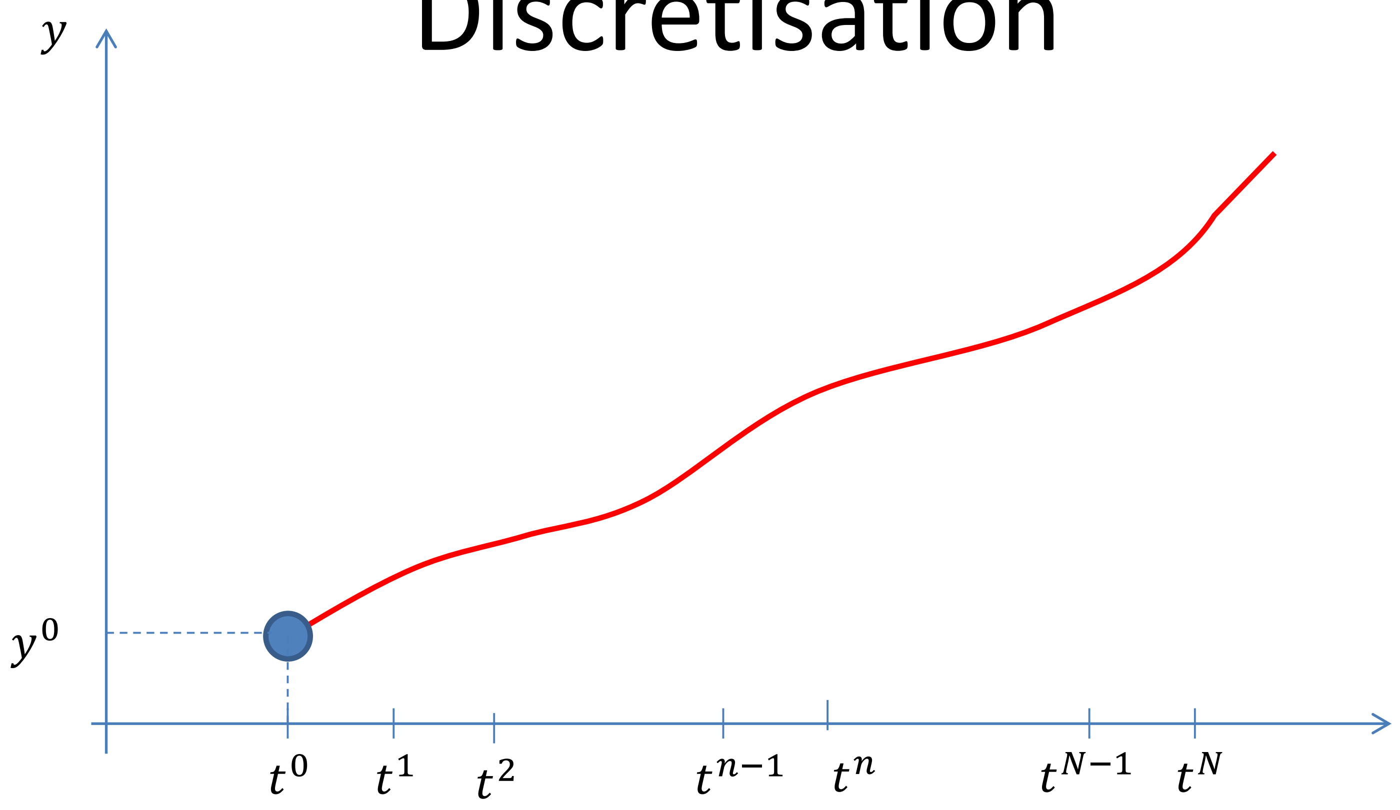
- Si une solution analytique ne peut pas être déterminée (c’est le cas en général …), une alternative est de résoudre l’équation de manière approchée, par simulation numérique

- Partant de la condition initiale $y(t^0) = y^0$, on utilise alors un algorithme qui génère une approximation de la solution $y(t)$ en un nombre fini d’instants:

$$t^n = t^0 + n\Delta t; \qquad\qquad y^n = y(t^0 + n\Delta t)$$

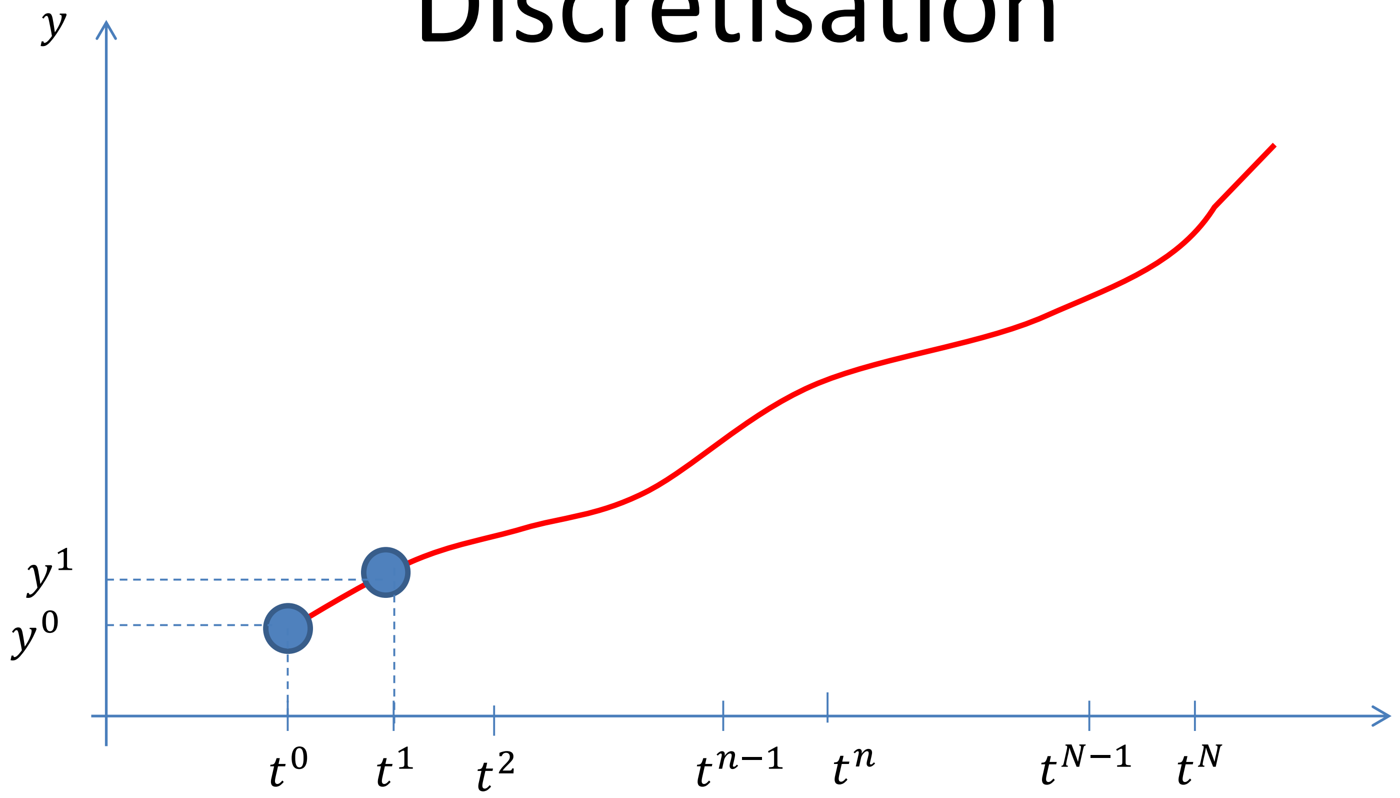
- Δt est le pas de temps de discrétisation (supposé constant dans le cadre de ce cours)

Discrétisation



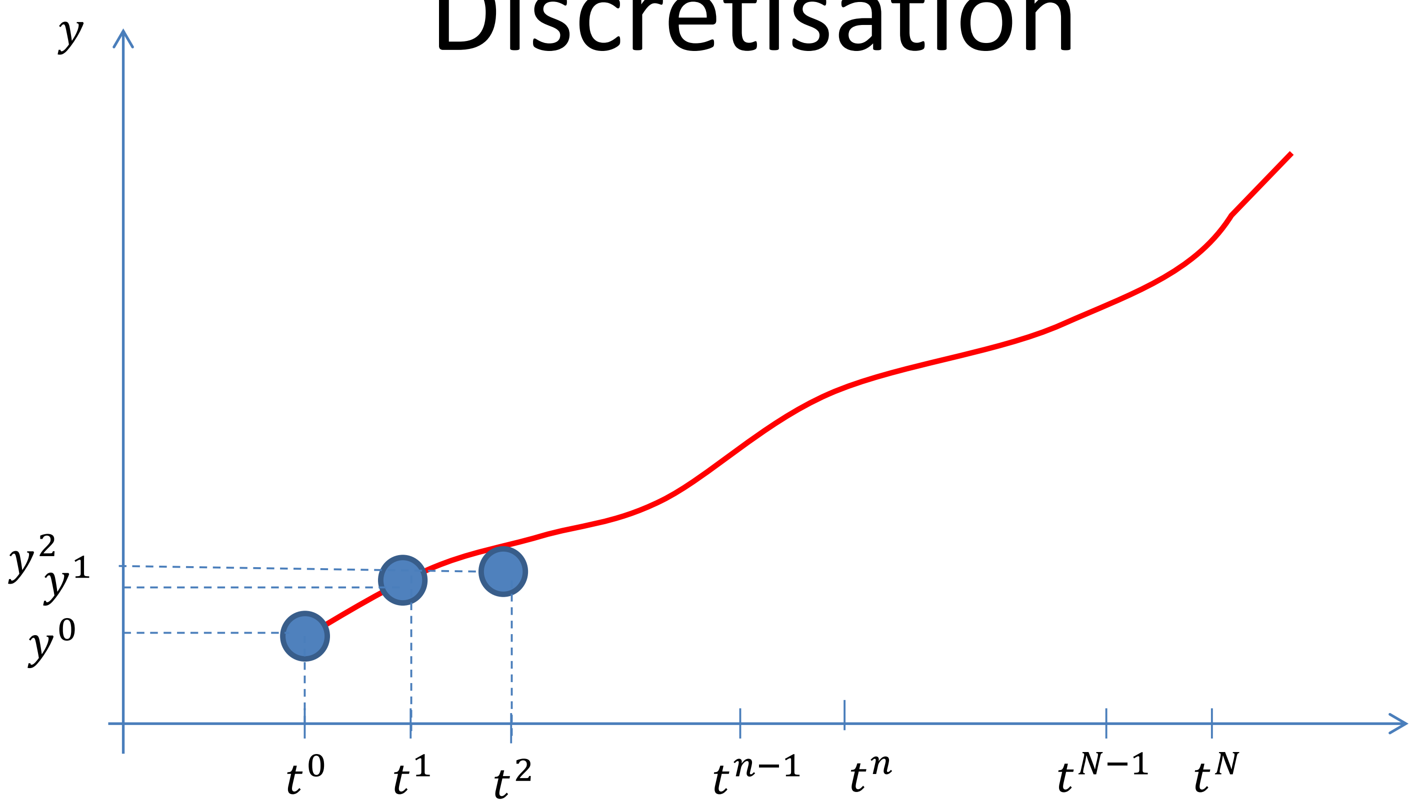
- Une fois le problème discrétisé, l'inconnue n'est plus **une fonction $\boldsymbol{y(t)}$** au sens mathématique mais **une suite de valeurs numériques** $y^0, y^1, \dots, y^{N-1}, y^N$ approchant les valeurs prises par $y(t)$ aux instants $t^0, t^1, \dots, t^{N-1}, t^N$

Discrétisation



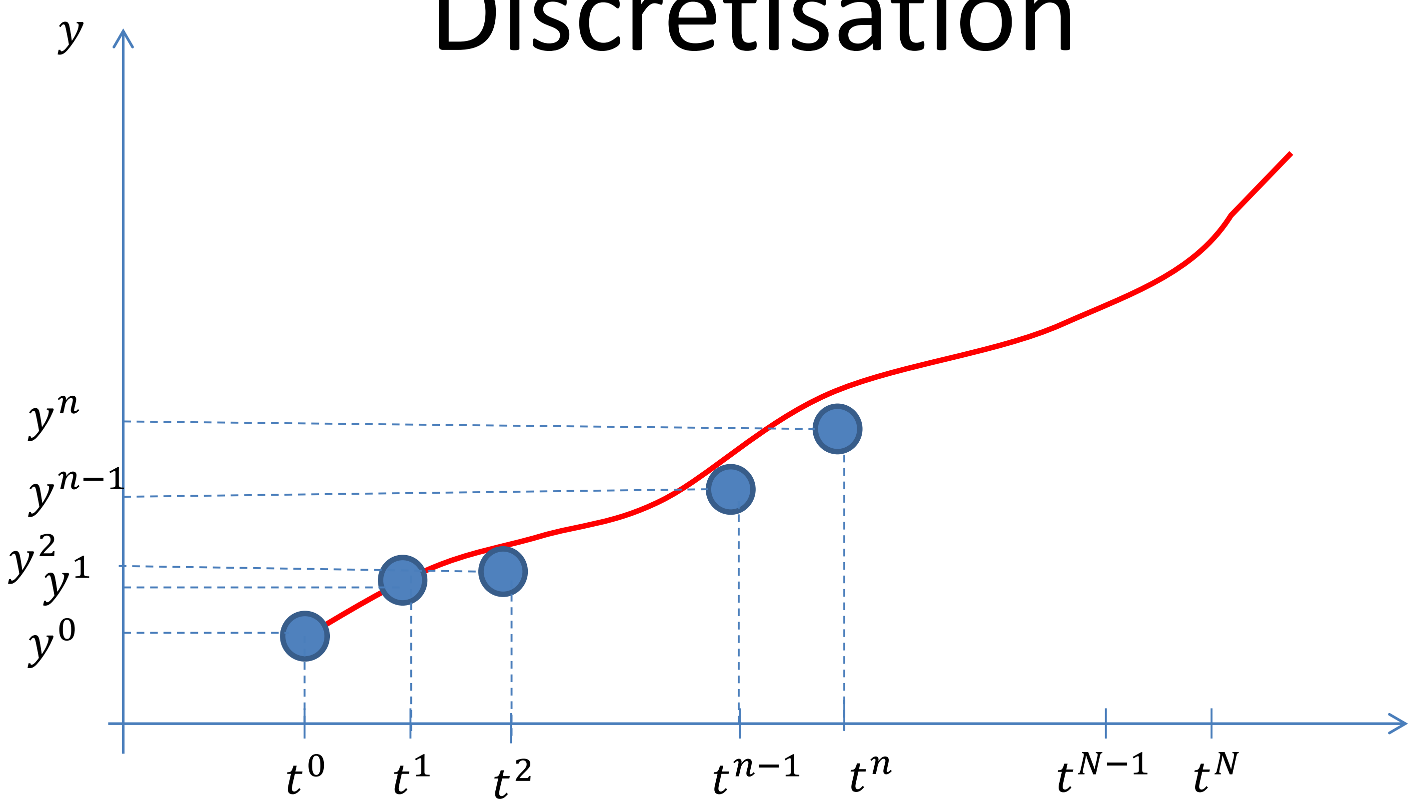
- Une fois le problème discrétisé, l'inconnue n'est plus **une fonction $\boldsymbol{y(t)}$** au sens mathématique mais **une suite de valeurs numériques** $y^0, y^1, \dots, y^{N-1}, y^N$ approchant les valeurs prises par $y(t)$ aux instants $t^0, t^1, \dots, t^{N-1}, t^N$

Discrétisation



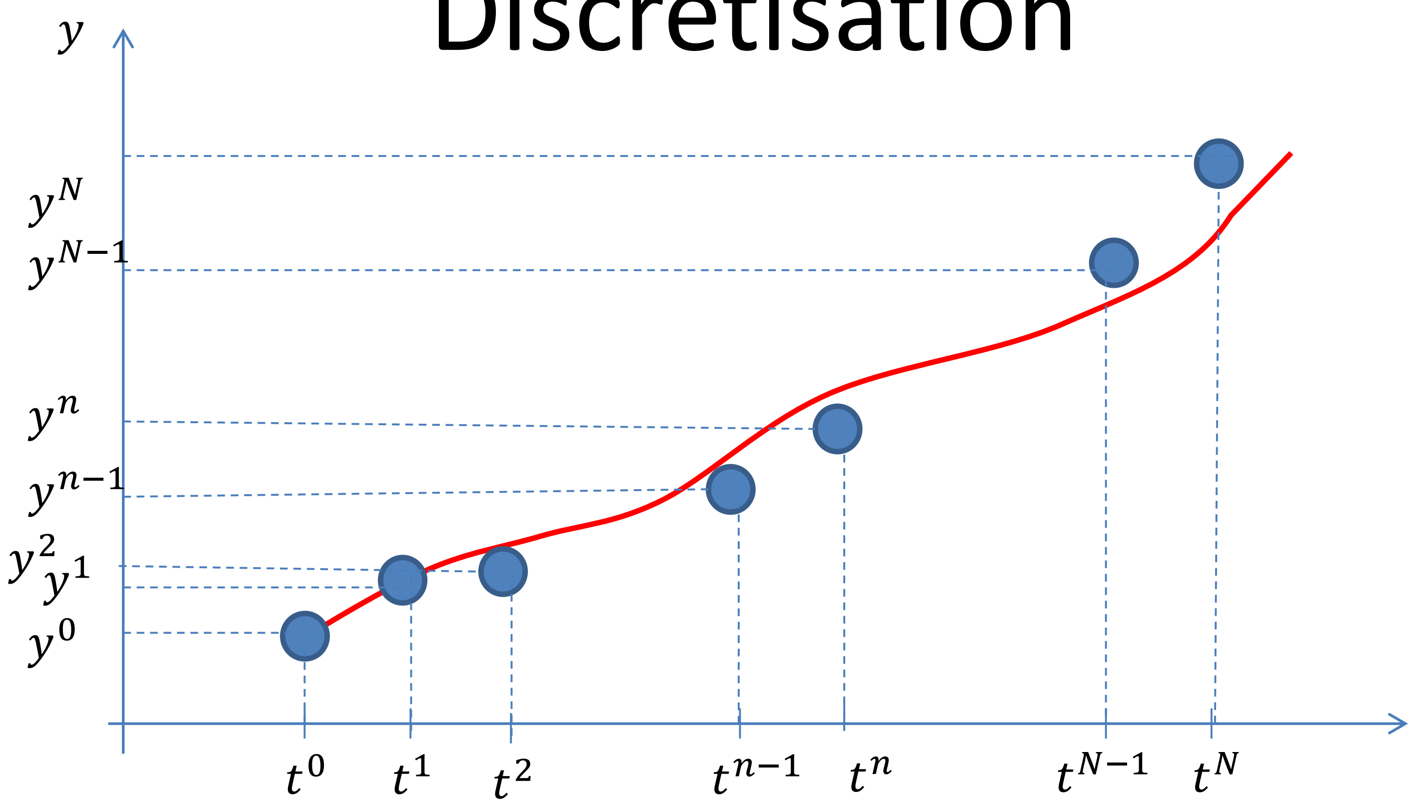
- Une fois le problème discrétisé, l'inconnue n'est plus **une fonction $\boldsymbol{y(t)}$** au sens mathématique mais **une suite de valeurs numériques** $y^0, y^1, \ldots, y^{N-1}, y^N$ approchant les valeurs prises par $y(t)$ aux instants $t^0, t^1, \ldots, t^{N-1}, t^N$

Discrétisation



- Une fois le problème discrétisé, l'inconnue n'est plus **une fonction $y(t)$** au sens mathématique mais **une suite de valeurs numériques** $y^0, y^1, \dots, y^{N-1}, y^N$ approchant les valeurs prises par $y(t)$ aux instants $t^0, t^1, \dots, t^{N-1}, t^N$

Discrétisation



- Une fois le problème discrétisé, l'inconnue n'est plus **une fonction $y(t)$** au sens mathématique mais **une suite de valeurs numériques** $y^0, y^1, \dots, y^{N-1}, y^N$ approchant les valeurs prises par $y(t)$ aux instants $t^0, t^1, \dots, t^{N-1}, t^N$

Quel algorithme ?

- Résoudre numériquement le problème

La fonction $\boldsymbol{F}$ étant donnée, trouver la fonction inconnue $\boldsymbol{y(t)}$ telle que $\boldsymbol{\frac{dy}{dt} = F(t,y) \quad \text{pour } t > t^0, \text{avec la cdt initiale } y(t^0) = y^0}$

revient donc à se doter d'un algorithme qui permet de progresser en temps et de générer y^{n+1} à partir des valeurs précédentes de y

- Toute la difficulté est de déterminer un « bon » algorithme

Algorithme exact

- La condition initiale fournissant la valeur y^0 de la fonction l'indice $n = 0$, un algorithme de résolution est en fait une relation de récurrence permettant de déterminer la valeur de la fonction inconnue à l'instant t^{n+1} connaissant ses valeurs aux instants précédents

$$y^0, y^1, \dots, y^{n-2}, y^{n-1}, y^n \quad \rightarrow \quad y^{n+1}$$

- Le point de départ est en général la relation exacte:

$$y^{n+1} = y^n + \int_{t^n}^{t^{n+1}} \frac{dy}{dt} dt = y^n + \int_{t^n}^{t^{n+1}} F(t, y) dt$$

Comment estimer l'intégrale de F ?

Méthodes à un pas

- En pratique, l'intégrale de F entre t^n et t^{n+1} n'est pas calculable de manière exacte car $y(t)$ n'est pas connue sur cet intervalle

- Les méthodes à un pas s'écrivent sous la forme:

$$\boxed{y^{n+1} = y^n + \Delta t \times \Phi}$$

- Φ est le taux de variation de y au cours de l'itération

- Puisque $\frac{dy}{dt} = F(t, y)$, sa valeur exacte est

$$\Phi_{exact} = \frac{y^{n+1} - y^n}{\Delta t} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} F(t, y) dt$$

ORDRE

- On cherche alors à approximer Φ à partir des valeurs disponibles de la fonction inconnue, soit $y^0, y^1, \dots, y^{n-1}, y^n, y^{n+1}$
- La méthode est alors d'ordre p si:

$$\boxed{\Phi_{exact} - \Phi = O\left(\Delta t^p\right)}$$

- L'erreur associée à une méthode diminue d'autant plus vite lorsque le pas de temps diminue que l'ordre est élevé:

Ordre (p)	0	1	2	4
Facteur de réduction de l'erreur suite à réduction de Δt d'un facteur 2	2^0=1	2^1=2	2^2=4	2^4=16

PRECISION DE LA SOLUTION ?

- La notion d'ordre est associé au taux de variation Φ. Mais ce qui nous intéresse c'est la **précision** avec laquelle la solution de l'équation est obtenue.
- Il faut donc comparer la solution approchée au temps physique T, soit $y^n = y(t = T = n\Delta t + t_0)$, à la solution exacte au même instant, soit $y_n^{ex} = y^{ex}(t = T)$. Or on obtient successivement:

$$y^n = y^n - y^{n-1} + y^{n-1} - y^{n-2} + \cdots + y^1 - y^0 + y^0$$

$$y^n = \Delta t \Phi_{n-1} + \Delta t \Phi_{n-2} + \cdots + \Delta t \Phi_0 + y^0$$

$$y^n = \Delta t((\Phi_{n-1}^{ex} + O(\Delta t^p)) + \Delta t((\Phi_{n-2}^{ex} + O(\Delta t^p)) + \cdots + \Delta t((\Phi_0^{ex} + O(\Delta t^p)) + y^0$$

$$y^n = \Delta t \Phi_{n-1}^{ex} + \Delta t \Phi_{n-2}^{ex} + \cdots + \Delta t \Phi_0^{ex} + y^0 + n\Delta t O(\Delta t^p)$$

$$y^n = y_n^{ex} - y_{n-1}^{ex} + y_{n-1}^{ex} - y_{n-2}^{ex} + \cdots + y_1^{ex} - y_0^{ex} + y^0 + T \times O(\Delta t^p)$$

$$\boldsymbol{y^n = y_n^{ex} + O(\Delta t^p)}$$

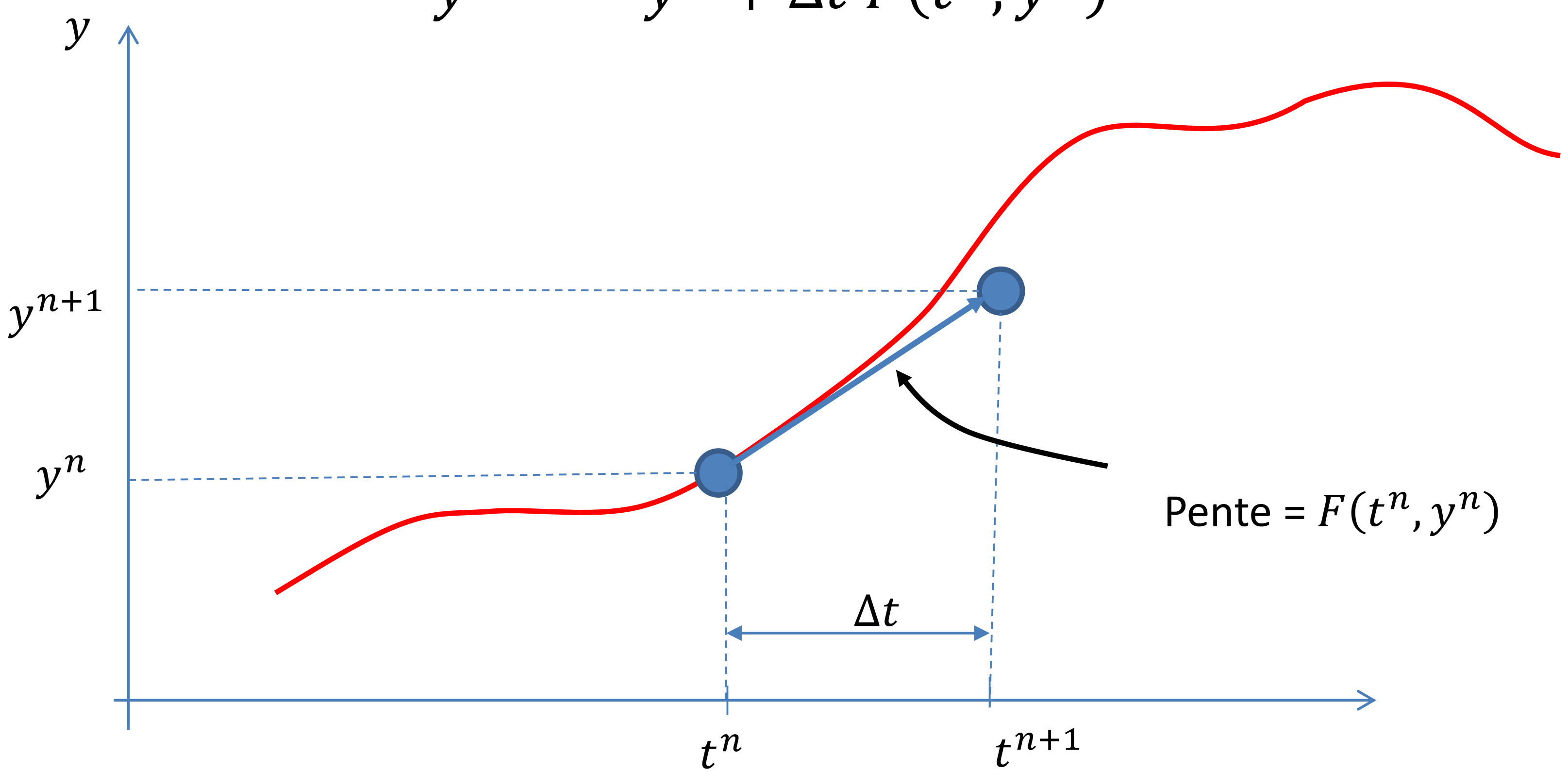
Méthodes à un pas classiques

- Euler explicite: $\Phi = F(t^n, y^n) = F^n$
$$y^{n+1} = y^n + \Delta t\, F(t^n, y^n)$$
- Euler implicite : $\Phi = F(t^{n+1}, y^{n+1}) = F^{n+1}$
$$y^{n+1} = y^n + \Delta t\, F(t^{n+1}, y^{n+1})$$
- Crank-Nicolson : $\Phi = \frac{1}{2}(F^n + F^{n+1})$
$$y^{n+1} = y^n + \frac{\Delta t}{2}(F^{n+1} + F^n)$$
- Adams-Bashforth: $\Phi = \frac{1}{2}(3F^n - F^{n-1})$
$$y^{n+1} = y^n + \frac{\Delta t}{2}(3F^n - F^{n-1})$$

Interprétation géométrique

Euler explicite

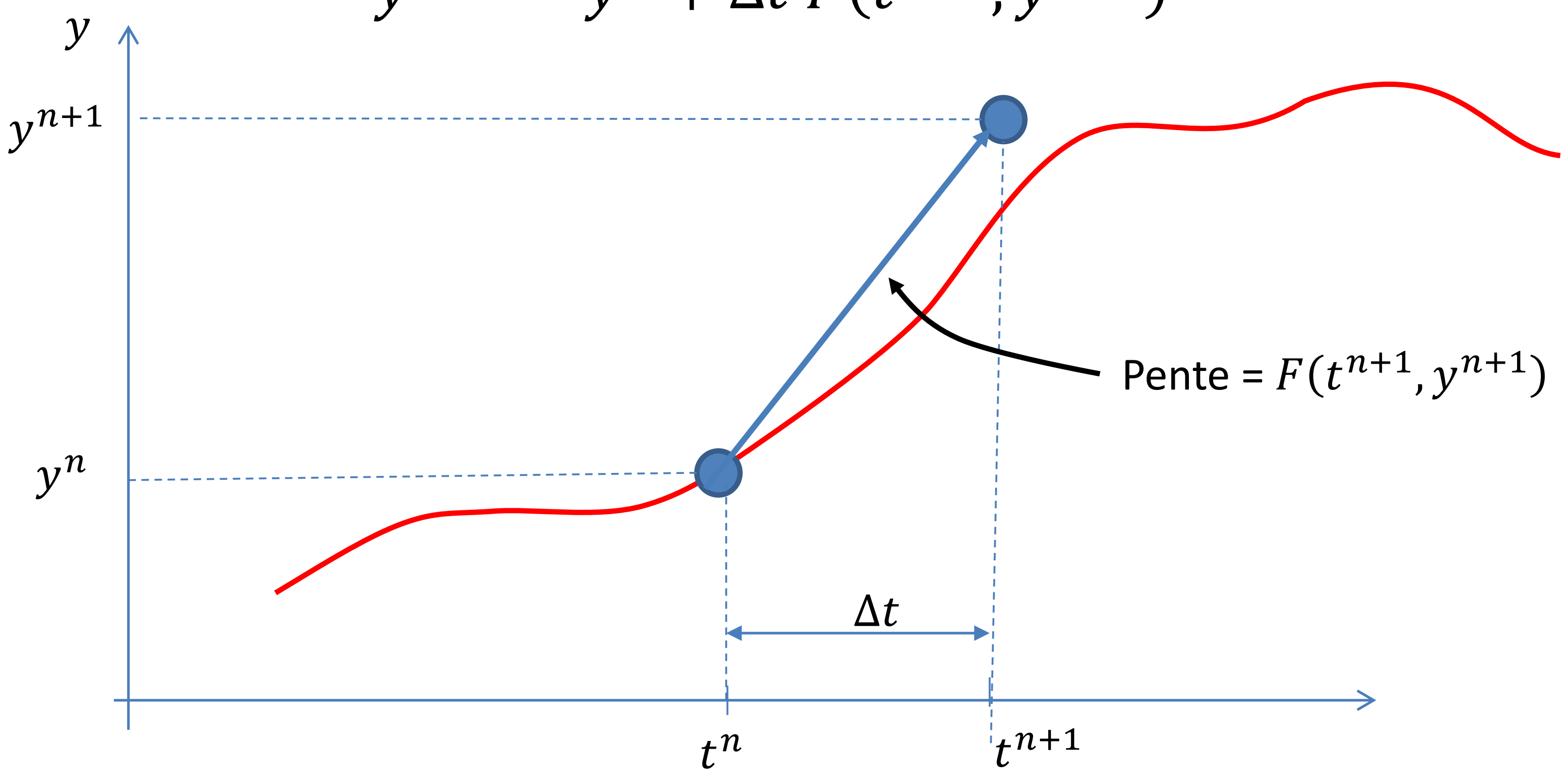
$$y^{n+1} = y^n + \Delta t\, F(t^n, y^n)$$



Interprétation géométrique

Euler implicite

$$y^{n+1} = y^n + \Delta t\, F(t^{n+1}, y^{n+1})$$



Calcul théorique de l'ordre

- L'ordre théorique d'une méthode s'obtient en caractérisant l'écart entre la valeur exacte de Φ et celle utilisée en pratique

  - Euler explicite : $\Phi = F^n$ → Ordre 1

  - Euler implicite : $\Phi = F^{n+1}$ → Ordre 1

  - Crank-Nicolson : $\Phi = \frac{1}{2}(F^n + F^{n+1})$ → Ordre 2

  - Adams-Bashforth : $\Phi = \frac{1}{2}(3F^n - F^{n-1})$ → Ordre 2

Augmentation de l'ordre

- On peut construire une approximation de y^{n+1} d'ordre arbitrairement élevé à partir du développement de Taylor suivant:

$$y^{n+1} = y^n + \frac{dy}{dt}(t^n)\Delta t + \frac{d^2y}{dt^2}(t^n)\frac{\Delta t^2}{2} + \ldots + \frac{d^py}{dt^p}(t^n)\frac{\Delta t^p}{p!} + O(\Delta t^{p+1})$$

- On peut conserver le cadre précédent des méthodes à un pas, $y^{n+1} = y^n + \Delta t \times \Phi$, à condition de définir Φ comme (avec un léger changement de notation , $\frac{d^iy}{dt^i}(t^n)$= $\left.\frac{d^iy}{dt^i}\right|^n$):

$$\Phi = \left.\frac{dy}{dt}\right|^n + \left.\frac{d^2y}{dt^2}\right|^n \frac{\Delta t}{2} + \ldots \left.\frac{d^py}{dt^p}\right|^n \frac{\Delta t^{p-1}}{p!} + O(\Delta t^p)$$

Augmentation de l'ordre

- En utilisant l'EDO vérifiée par y et des règles de dérivation usuelles:

$$\left.\frac{dy}{dt}\right|^n = F(t^n, y^n) = F^n$$

$$\left.\frac{d^2y}{dt^2}\right|^n = \frac{dF}{dt}(t^n, y^n) = \left[\frac{\partial F}{\partial t} + \frac{\partial F}{\partial y}\frac{dy}{dt}\right]^n = F_t^n + F_y^n F^n$$

$$\left.\frac{d^3y}{dt^3}\right|^n = F_{tt}^n + 2F_{ty}^n F^n + F_{yy}^n {F_y^n}^2 + F_y^n\left(F_t^n + F_y^n F^n\right)$$

- Et donc:

$$\Phi = F^n + \left[F_t^n + F_y^n F^n\right]\frac{\Delta t}{2}$$
$$+ \left[F_{tt}^n + 2F_{ty}^n F^n + F_{yy}^n {F_y^n}^2 + F_y^n\left(F_t^n + F_y^n F^n\right)\right]\frac{\Delta t^2}{6} + \cdots$$

Méthodes à pas multiples

- Les développements précédents ne sont pas ou peu utilisables en pratique car ils font intervenir des dérivées d'ordres élevées de la fonction F.

- Or, cette fonction n'est pas nécessairement connue de manière analytique et calculer ses dérivées successives peut s'avérer coûteux et imprécis

- Plutôt que d'estimer des dérivées d'ordre élevées, il est préférable de mieux choisir les endroits où on évalue F.

- Illustration avec le développement de Taylor à l'ordre 2 qui donne

$$\Phi = F^n + \left[F_t^n + F_y^n F^n\right] \frac{\Delta t}{2}$$

Méthodes à pas multiples: ordre 2

- On cherche à approximer $\Phi = F^n + \left[F_t^n + F_y^n F^n\right]\frac{\Delta t}{2}$ à l'ordre 1 en Δt à l'aide de

$$\bar{\Phi} = A_1 \underbrace{F(t^n, y^n)}_{F^n} + A_2 F(t^n + \alpha\Delta t, y^n + \beta\Delta t F^n)$$

où A_1, A_2, α et β sont des paramètres à déterminer

- Par identification jusqu'à l'ordre 1 inclus on obtient:

$$F^n + \left[F_t^n + F_y^n F^n\right]\frac{\Delta t}{2} = (A_1 + A_2)F^n + A_2\left(\alpha\Delta t \times F_t^n + \beta\Delta t F^n F_y^n\right)$$

d'où l'on tire: $A_2\alpha = \frac{1}{2}$ $A_2\beta = \frac{1}{2}$ $A_1 + A_2 = 1$

Runge-Kutta d'ordre 2

- Une fois l'identification réalisée, on peut proposer l'approximation $y^{n+1} = y^n + \Delta t \times \overline{\Phi}$ pour y^{n+1}
- On obtient en fait une famille de schémas (3 contraintes mais 4 paramètres à déterminer). Pour le choix particulier $A_1 = 0$, on obtient:

$$y^{n+1} = y^n + \Delta t \times F\left(t^n + \frac{\Delta t}{2}, y^n + \frac{k_1 \Delta t}{2}\right)$$

- Ce n'est pas une méthode à un pas. En effet, cet algorithme nécessite le calcul successif de deux valeurs prises par la fonction F
    - Etape 1: calcul de $k_1 = F(t^n, y^n)$
    - Etape 2: calcul de $k_2 = F\left(t^n + \frac{\Delta t}{2}, y^n + \frac{k_1 \Delta t}{2}\right)$

On en déduit alors l'approximation de la fonction $y(t)$ à l'instant t^{n+1} :

$$y^{n+1} = y^n + \Delta t \times k_2$$

Runge-Kutta ordre 3

- On peut reprendre la même démarche en conservant un terme de plus dans le développement de Taylor.
- A l'ordre 3 on obtient la méthode à 3 pas suivante:

$$k_1 = F(t^n, y^n)$$
$$k_2 = F\left(t^n + \frac{\Delta t}{2}, y^n + \frac{k_1 \Delta t}{2}\right)$$
$$k_3 = F(t^n + \Delta t, y^n - k_1 \Delta t + 2k_2 \Delta t)$$

$$y^{n+1} = y^n + \frac{\Delta t}{6} \times (k_1 + 4k_2 + k_3)$$

Runge-Kutta ordre 4

- On peut reprendre la même démarche en conservant deux termes de plus dans le développement de Taylor.
- A l'ordre 4 on obtient la méthode à 4 pas suivante:

$$k_1 = F(t^n, y^n)$$
$$k_2 = F\left(t^n + \frac{\Delta t}{2}, y^n + \frac{k_1 \Delta t}{2}\right)$$
$$k_3 = F\left(t^n + \frac{\Delta t}{2}, y^n + \frac{k_2 \Delta t}{2}\right)$$
$$k_4 = F(t^n + \Delta t, y^n + k_3 \Delta t)$$

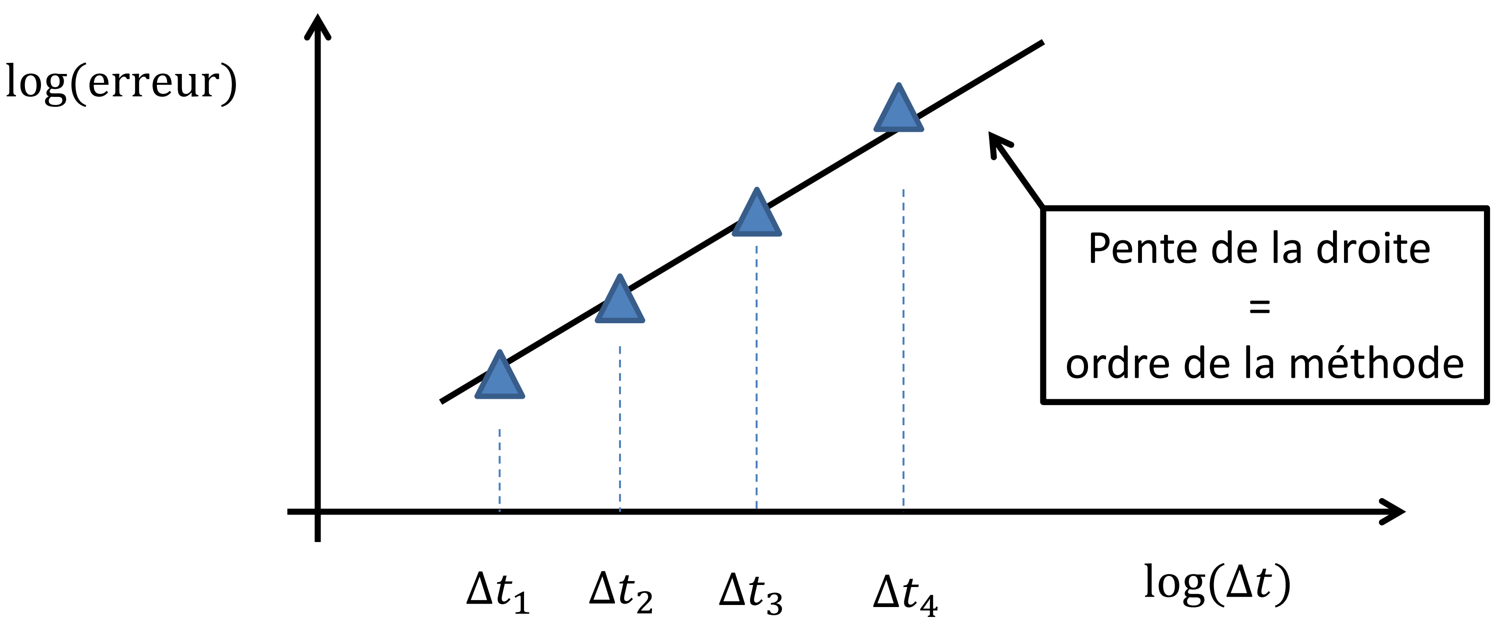
$$y^{n+1} = y^n + \frac{\Delta t}{6} \times (k_1 + 2k_2 + 2k_3 + k_4)$$

Mesure numérique de l’ordre

- Lorsque l’on utilise un logiciel de résolution d’une équation différentielle, il est utile de pouvoir vérifier ses performances
- Il est possible de « mesurer » à partir d’une expérience numérique l’ordre effectif de la méthode implémentée
- Partant de la définition de l’ordre p, on peut déduire que $y^n - y_n^{ex} = O(\Delta t^p) \approx \alpha \Delta t^p$ où α est une constante
- On en déduit que $\log|y^n - y_n^{ex}| \approx \log\alpha + p \log \Delta t$ lorsque $\Delta t \rightarrow 0$
- L’ordre effectif d’une méthode peut donc être mesuré en traçant dans un diagramme logarithmique l’erreur obtenue pour un problème donné et pour différentes valeurs du pas de temps; **la pente de la droite obtenue est égale à l’ordre recherché**.

Mesure numérique de l’ordre

- Pour mesurer l’ordre d’une méthode, il faut réaliser plusieurs simulations identiques (même problème résolu) avec des pas de temps différents



Relation précision-coût

- Le coût d'une itération dépend du nombre de fois que de la fonction F doit être calculée: n fois pour une méthode d'ordre n
- Ne pas déduire qu'une méthode est d'autant plus lente que son ordre est élevé; tout dépend de la précision désirée …
- La précision augmente lorsque l'erreur diminue
- Le coût total augmente lorsque le pas de temps Δt diminue