1 D'où ça vient (linéaire)

- diffusion implicite

$$\frac{\partial c}{\partial t} = \frac{1}{\rho}\frac{\partial}{\partial z}\left(\rho K \frac{\partial c}{\partial z}\right)$$
$$c^{n+1} - \frac{\tau}{\rho}\frac{\partial}{\partial z}\left(\rho K \frac{\partial c^{n+1}}{\partial z}\right) = c^n$$

- SW semi-implicite

$$\begin{aligned}
\frac{\partial u}{\partial t} - (f+\zeta)v + \frac{\partial}{\partial x}\left(\frac{u^2+v^2}{2} + gh\right) &= 0 \\
\frac{\partial v}{\partial t} + (f+\zeta)u + \frac{\partial}{\partial y}\left(\frac{u^2+v^2}{2} + gh\right) &= 0 \\
\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}\left((H+h)\,u\right) + \frac{\partial}{\partial y}\left((H+h)\,v\right) &= 0
\end{aligned}$$

$$\begin{aligned}
\frac{u^{n+1}-u^n}{\tau} - (f+\zeta^n)v^n + \frac{\partial}{\partial x}\left(K^n + gh^{n+1}\right) &= 0 \\
\frac{v^{n+1}-v^n}{\tau} + (f+\zeta)u^n + \frac{\partial}{\partial y}\left(K^n + gh^{n+1}\right) &= 0 \\
\frac{\partial h^{n+1}}{\partial t} + \frac{\partial}{\partial x}\left(Hu^{n+1} + h^n u^n\right) + \frac{\partial}{\partial y}\left(Hv^{n+1} + h^n v^n\right) &= 0 \\
\left[1 - \frac{\partial}{\partial x}\left(gH\tau^2\frac{\partial}{\partial x}\right) - \frac{\partial}{\partial y}\left(gH\tau^2\frac{\partial}{\partial y}\right)\right] h^{n+1} &= F[u^n, v^n, h^n]
\end{aligned}$$

- QG

$$q = \Delta\psi - \frac{f^2}{gH}\psi$$
$$\Leftrightarrow \frac{gH}{f^2}\psi - \psi = -\frac{gH}{f^2}q$$

- générique

$$\begin{aligned}
A\psi - \operatorname{div}\left(B\nabla\psi\right) =& rhs \\
Q\left[\psi\right] =& \frac{1}{2}\int A\psi^2 + B\left(\nabla\psi\right)^2 \mathrm{d}x \\
Q\left[\psi^{n+1} + \varepsilon\psi'\right] =& Q\left[\psi^{n+1}\right] + \varepsilon^2 Q\left[\psi'\right] \\
&+ \varepsilon\int \psi'\left(A - \operatorname{div}B\nabla\right)\psi^{n+1}\mathrm{d}x + bord \\
=& Q\left[\psi^{n+1}\right] + \varepsilon^2 Q\left[\psi'\right]
\end{aligned}$$

2 Résoudre

- 1D => tridiagonal => OK

$$\begin{aligned}
\delta^2 A_i\psi_i + B_{i-1/2}\left(\psi_i - \psi_{i-1}\right) - B_{i+1/2}\left(\psi_{i+1} - \psi_i\right) &= R_i \\
-B_{i+1/2}\psi_{i-1} + \left(\delta^2 A_i + B_{i-1/2} + B_{i+1/2}\right)\psi_i - B_{i+1/2}\psi_{i+1} &= R_i
\end{aligned}$$

$$\Updownarrow$$
$$\begin{aligned}
&\left(L\psi\right)_i = R_i \\
&\left(L\psi\right)_i \equiv \delta^2 A_i\psi_i + B_{i-1/2}\left(\psi_i - \psi_{i-1}\right) - B_{i+1/2}\left(\psi_{i+1} - \psi_i\right)
\end{aligned}$$

- 2D coeff constants => Fourier (FFT= Fast Fourier Transforme coûte $O(N \log N)$.

$$\begin{aligned}
x &\in [0, L] \\
x_i &= \frac{i+1/2}{N} L \\
\psi_i &= \sum_k \hat{\psi}_k \cos k \frac{\pi x_i}{L} \qquad k = 0 \dots N \\
B\left(\psi_{i-1} + \psi_{i+1}\right) &= 2B \sum_k \hat{\psi}_k \cos k \frac{\pi x_i}{L} \cos \frac{k\pi}{N} \\
\delta^2 A + 2B \left(1 - \cos \frac{k\pi}{N}\right) \hat{\psi}_k &= \hat{R}_i
\end{aligned}$$

- général :
  - weighted Jacobi

$$\begin{aligned}
L &= D + (L - D) \\
(D\psi)_i &= \left(\delta^2 A + 2B\right) \psi_i \\
((L-D)\psi)_i &= B\left(\psi_{i-1} + \psi_{i+1}\right)
\end{aligned}$$

$$\begin{aligned}
D\psi &= (D-L)\psi + R \\
D\psi^{(m+1)} &= (D-L)\psi^{(m)} + R \\
\psi^{(m+1)} &= (1-\omega)\psi^{(m)} + \omega\left((1 - D^{-1}L)\psi^{(m)} + D^{-1}R\right) \\
(1 - D^{-1}L)\hat{\psi}_k &= \frac{2B\cos\frac{k\pi}{N}}{\delta^2 A + 2B} \hat{\psi}_k \\
\hat{\psi}_k^{(m+1)} - \hat{\psi}_k^{(m)} &= \alpha(k)\left(\hat{\psi}_k^{(m)} - \hat{\psi}_k^{(m-1)}\right) \\
\alpha(k) &= (1-\omega) + \omega \frac{2B\cos\frac{k\pi}{N}}{\delta^2 A + 2B} \\
\alpha(k=0) &= (1-\omega) + \omega\left(1 + \frac{\delta^2 A}{2B}\right)^{-1} \\
&\simeq 1 - \omega \frac{\delta^2 A}{2B} \qquad (\delta^2 A \ll B) \\
\alpha(k=N) &= (1-\omega) - \omega \frac{2B\cos\frac{k\pi}{N}}{\delta^2 A + 2B} \\
\alpha(k=N/2) &= (1-\omega)
\end{aligned}$$

$k = N/2$

$$\begin{aligned}
0 = \frac{\alpha(N/2) + \alpha(N)}{2} = 1 - \omega + \omega \frac{B}{\delta^2 A + 2B} &= 0 \\
\omega\left(1 - \frac{B}{\delta^2 A + 2B}\right) &= 1 \\
\omega &= \frac{\delta^2 A + 2B}{\delta^2 A + B} \\
1 - \omega &= \frac{\delta^2 A + 2B}{\delta^2 A + 4B}
\end{aligned}$$

$$\hat{\psi}_k^{(m+1)} - \hat{\psi}_k^{(m)} = \frac{2B}{\delta^2 A + 4B}\left(1 + \cos\frac{k\pi}{N}\right)\left(\hat{\psi}_k^{(m)} - \hat{\psi}_k^{(m-1)}\right)$$

– convergence rapide pour $k \geq N/2$, la plus lente pour $k = 0$:

$$\hat{\psi}_k^{(m)} - \hat{\psi}_k^{(\infty)} = \left(1 + \frac{\delta^2 A}{4B}\right)^{-m} \left(\hat{\psi}_k^{(0)} - \hat{\psi}_k^{(\infty)}\right)$$

$$m = \frac{-\log \varepsilon}{\log\left(1 + \frac{\delta^2 A}{4B}\right)} \sim -\log \varepsilon \frac{B}{\delta^2 A}$$

=> nombre de conditionnement
QG : $A = 1, \quad B = R_d^{-2}$

$$\left(1 + \frac{\delta^2}{4R_d^2}\right)^{-m} \qquad m \sim \delta^{-2} R_d^{-2}$$

SW semi-implicite :

$$\left(1 + \frac{\delta^2}{gH\tau^2}\right)^{-m} m \sim C^2 \qquad C = \frac{\tau\sqrt{gH}}{\delta}$$

⇒ pas utile en pratique

– CG/PCG
– MG

3 Non-linéaire

- Point fixe
- Newton-Raphson
- Quasi-Newton

https://en.wikipedia.org/wiki/Conjugate_gradient_method

Conjugate gradient method

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

$\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$

$\mathbf{p}_0 := \mathbf{r}_0$

$k := 0$

repeat

$$\alpha_k := \frac{\mathbf{r}_k^\mathrm{T}\mathbf{r}_k}{\mathbf{p}_k^\mathrm{T}\mathbf{A}\mathbf{p}_k}$$

$\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k\mathbf{p}_k$

$\mathbf{r}_{k+1} := \mathbf{r}_k - \alpha_k\mathbf{A}\mathbf{p}_k$

if $\mathbf{r}_{k+1}$ is sufficiently small **then** exit loop **end if**

$$\beta_k := \frac{\mathbf{r}_{k+1}^\mathrm{T}\mathbf{r}_{k+1}}{\mathbf{r}_k^\mathrm{T}\mathbf{r}_k}$$

$\mathbf{p}_{k+1} := \mathbf{r}_{k+1} + \beta_k\mathbf{p}_k$

$k := k + 1$

end repeat

$$\|x^* - x_k\|_\mathbf{A} \leqslant 2\left(\frac{\sqrt{\kappa(\mathbf{A})}-1}{\sqrt{\kappa(\mathbf{A})}+1}\right)^k \|x^* - x_0\|_\mathbf{A}$$

Note, the important limit when $\kappa(\mathbf{A})$ tends to ∞

$$\frac{\sqrt{\kappa(\mathbf{A})}-1}{\sqrt{\kappa(\mathbf{A})}+1} \approx 1 - \frac{2}{\sqrt{\kappa(\mathbf{A})}} \quad \text{for} \quad \kappa(\mathbf{A}) \gg 1\,.$$

This limit shows a faster convergence rate compared to the iterative methods of Jacobi or Gauss-Seidel which scale as $\approx 1 - \frac{2}{\kappa(\mathbf{A})}$.

https://en.wikipedia.org/wiki/Conjugate_gradient_method

Preconditioned conjugate gradient method (PCG)

$$\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b}$$

$\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$

$\mathbf{z}_0 := \mathbf{M}^{-1}\mathbf{r}_0$

$\mathbf{p}_0 := \mathbf{z}_0$

$k := 0$

repeat

$$\alpha_k := \frac{\mathbf{r}_k^\mathsf{T}\mathbf{z}_k}{\mathbf{p}_k^\mathsf{T}\mathbf{A}\mathbf{p}_k}$$

$\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{p}_k$

$\mathbf{r}_{k+1} := \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k$

if $\mathbf{r}_{k+1}$ is sufficiently small **then** exit loop **end if**

$\mathbf{z}_{k+1} := \mathbf{M}^{-1}\mathbf{r}_{k+1}$

$$\beta_k := \frac{\mathbf{z}_{k+1}^\mathsf{T}\mathbf{r}_{k+1}}{\mathbf{z}_k^\mathsf{T}\mathbf{r}_k}$$

$\mathbf{p}_{k+1} := \mathbf{z}_{k+1} + \beta_k \mathbf{p}_k$

$k := k + 1$

end repeat

The above formulation is equivalent to applying the conjugate gradient method without preconditioning to the system[1]

$$\mathbf{E}^{-1}\mathbf{A}(\mathbf{E}^{-1})^\mathsf{T}\hat{\mathbf{x}} = \mathbf{E}^{-1}\mathbf{b}$$

where

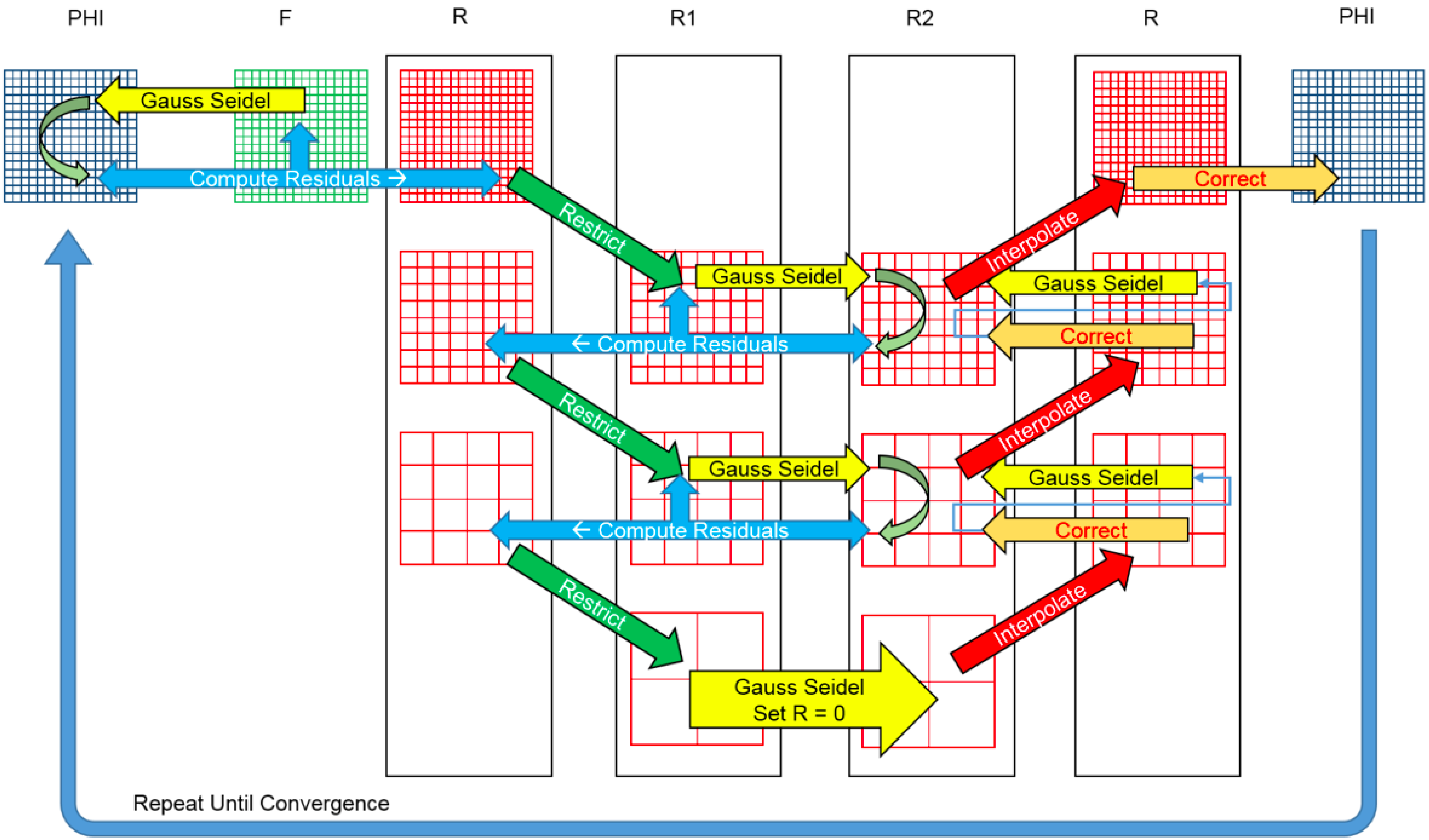
$$\mathbf{E}\mathbf{E}^\mathsf{T} = \mathbf{M}, \qquad \hat{\mathbf{x}} = \mathbf{E}^\mathsf{T}\mathbf{x}.$$

$\Rightarrow$ works if $\mathbf{M} \simeq \mathbf{A}$ and M^{-1} is cheap to compute.

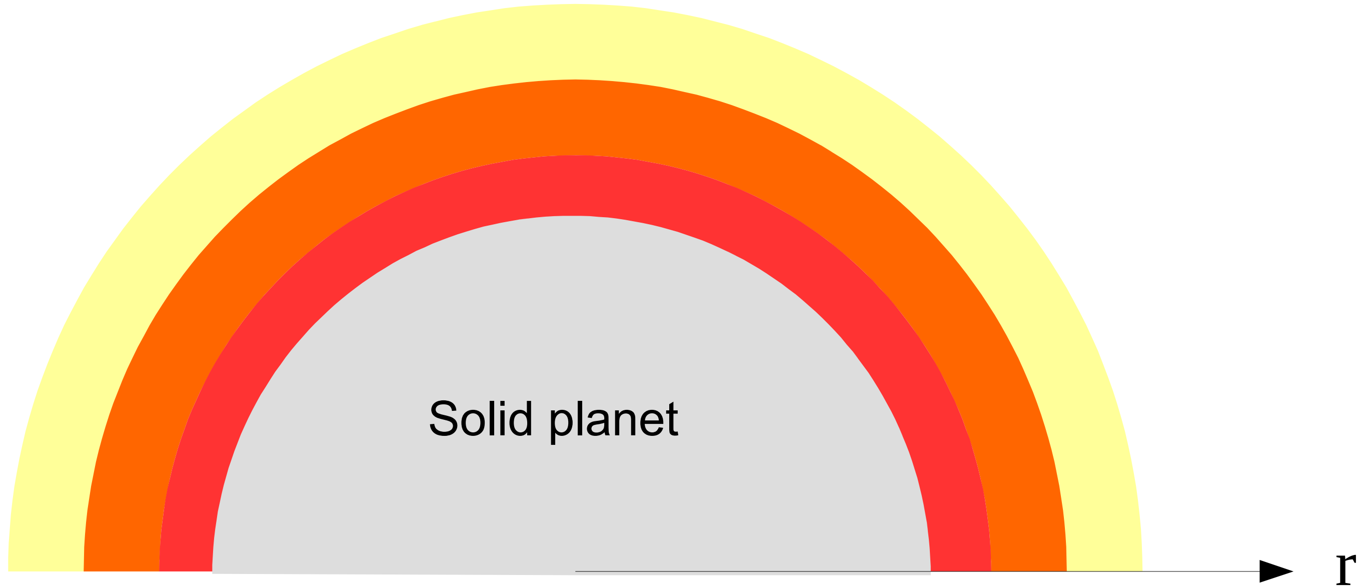
Example :

- FFT solver costs O(Nlog(N)) for N degrees of freedom, but limited to constant coefficients and simple domain shape (sphere/rectangle)
- Multigrid solver costs O(N)

Multigrid V-Cycle: Solving **PHI** in PDE ***f*** **(PHI) = F**



Hydrostatic adjustment



Initial condition not in hydrostatic balance => vertical adiabatic motion until balance is reached

balance <=> minimum of total enthalpy

$$E[\Phi] = \int [e(\alpha,\theta) + \Phi]\, m(\eta)\mathrm{d}\eta + p_\infty V(\Phi(\eta = 1))$$

$$e = C_v T = C_v \theta \left(\frac{R\theta}{\alpha p_r}\right)^{R/C_v}$$

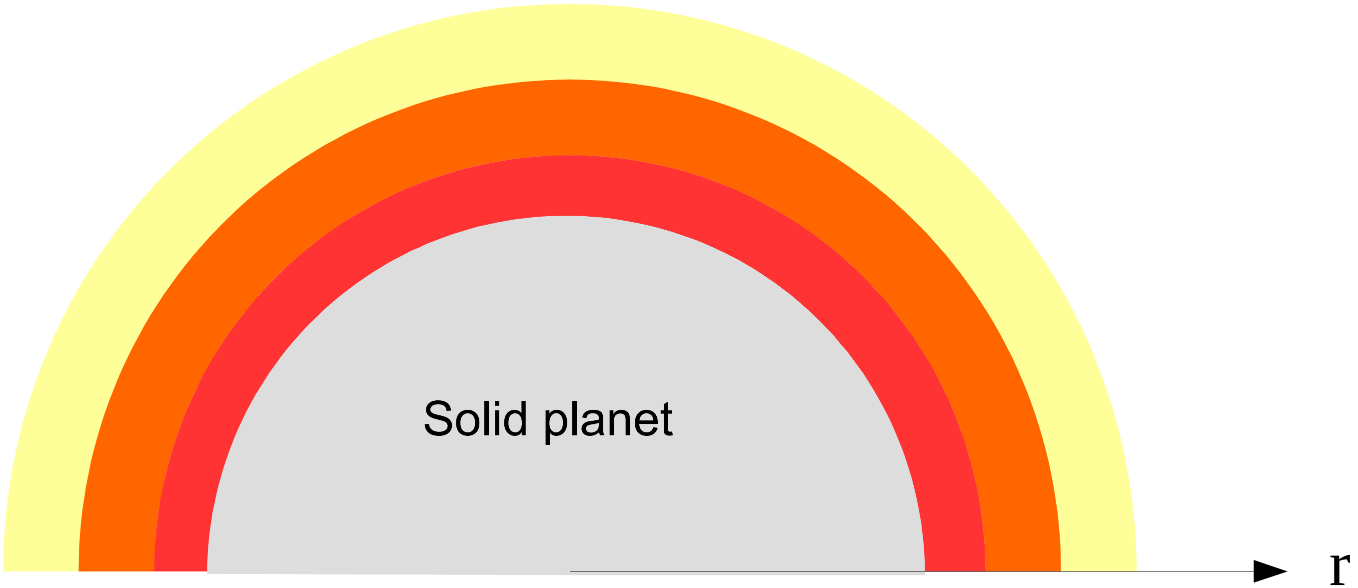
$$\alpha = \frac{1}{\rho} = \frac{1}{m}\frac{\mathrm{d}}{\mathrm{d}\eta} V(\Phi)$$

$$\Phi = g_0 a \left(1 - \frac{a}{r}\right)$$

$$g = \frac{\partial \Phi}{\partial r} = g_0 \frac{a^2}{r^2}$$

$$V = \frac{4}{3}\pi r^3 = \frac{4}{3}\pi a^3 \left(1 - \frac{\Phi}{g_0 a}\right)^{-3}$$

Hydrostatic adjustment

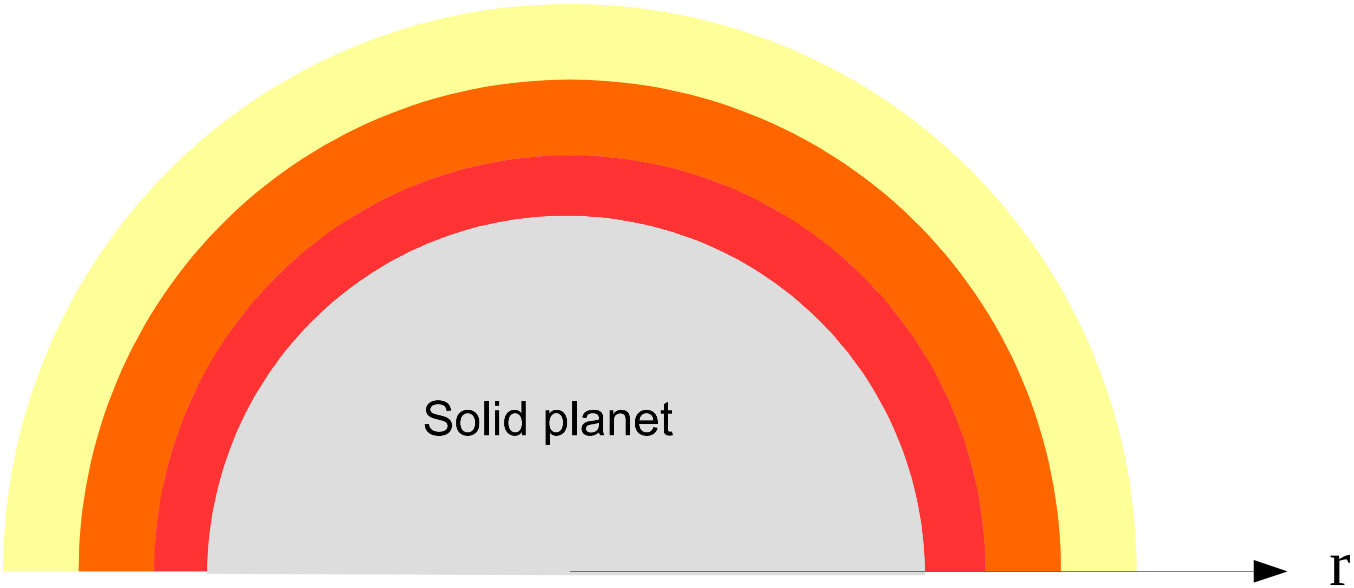


balance <=> minimum of total enthalpy

$$E(\Phi_1,\dots\Phi_N) = \sum_{k=1}^{N}\left[e\left(\frac{V(\Phi_k)-V(\Phi_{k-1})}{m_k},\theta_k\right)+\frac{\Phi_{k-1}+\Phi_k}{2}\right]m_k + p_\infty V(\Phi_N),$$

$$e(\alpha,\theta) = C_v\theta\left(\frac{R\theta}{\alpha p_r}\right)^{R/C_v} \qquad V = \frac{4}{3}\pi a^3\left(1-\frac{\Phi}{g_0 a}\right)^{-3}$$

Hydrostatic adjustment



balance <=> minimum of total enthalpy

$$E(\Phi_1, \dots \Phi_N) = \sum_{k=1}^{N} \left[e\left(\frac{V(\Phi_k) - V(\Phi_{k-1})}{m_k}, \theta_k \right) + \frac{\Phi_{k-1} + \Phi_k}{2} \right] m_k + p_\infty V(\Phi_N),$$

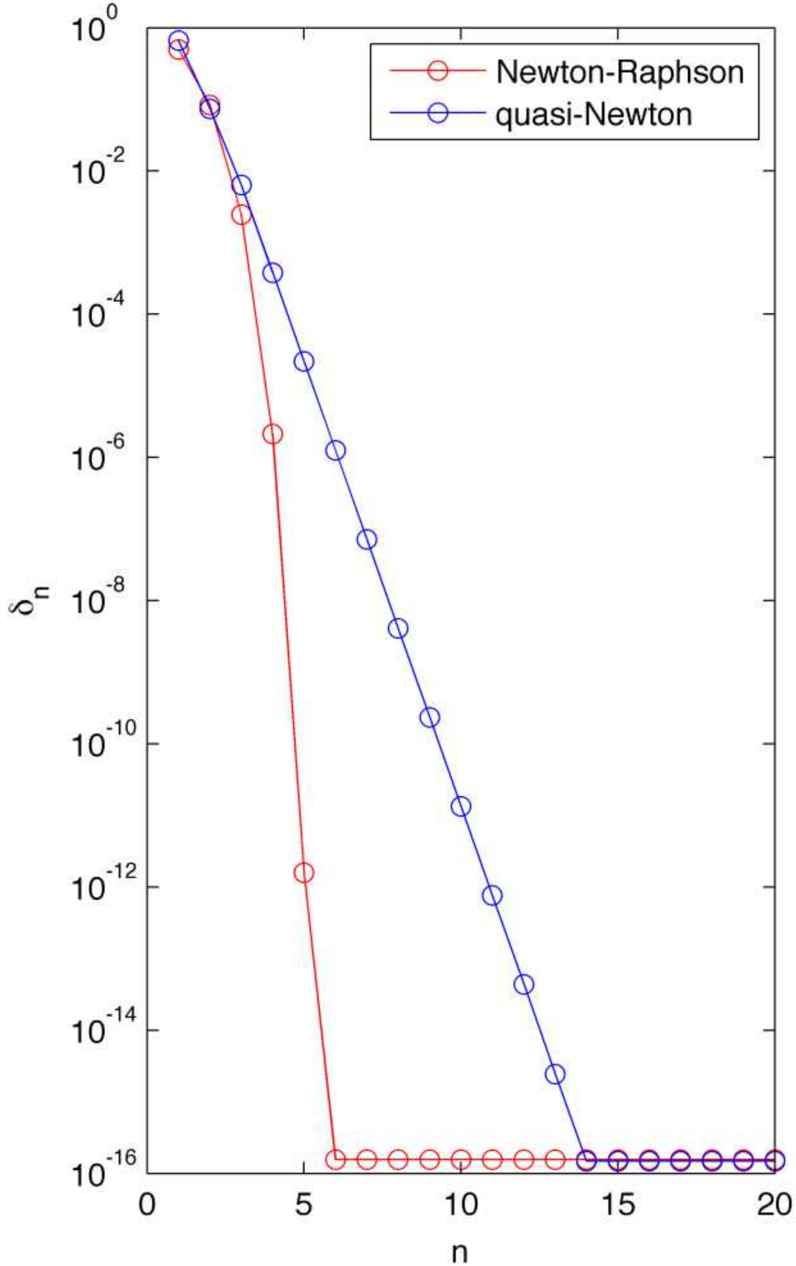
$$(k > N) \qquad f_k = \frac{\partial E}{\partial \Phi_k} = \frac{m_k + m_{k+1}}{2} + (p_{k+1} - p_k) \frac{\mathrm{d}V}{\mathrm{d}\Phi}(\Phi_k)$$

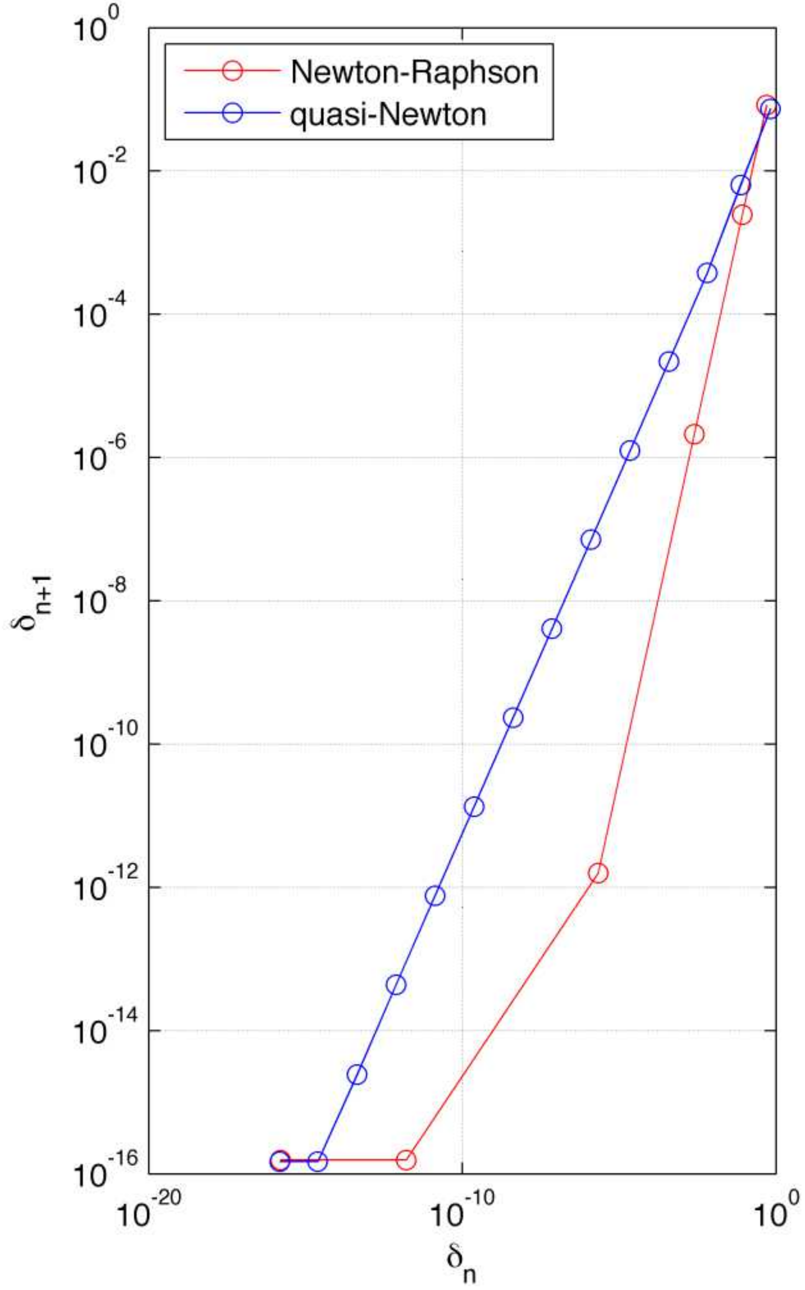
$$f_N = \frac{\partial E}{\partial \Phi_N} = \frac{m_N}{2} + (p_\infty - p_N) \frac{\mathrm{d}V}{\mathrm{d}\Phi}(\Phi_N)$$

$$\alpha_k = \frac{V(\Phi_k) - V(\Phi_{k-1})}{m_k}$$

$$p_k = -\frac{\partial e}{\partial \alpha}(\alpha_k, \theta_k)$$

Fixed-point versus Newton-Raphson





Fixed-point versus Newton-Raphson

Fixed point :

$$\delta_{n+1} = c.\delta_n \quad \Rightarrow \quad \log(\delta_n) = n \log(c)$$

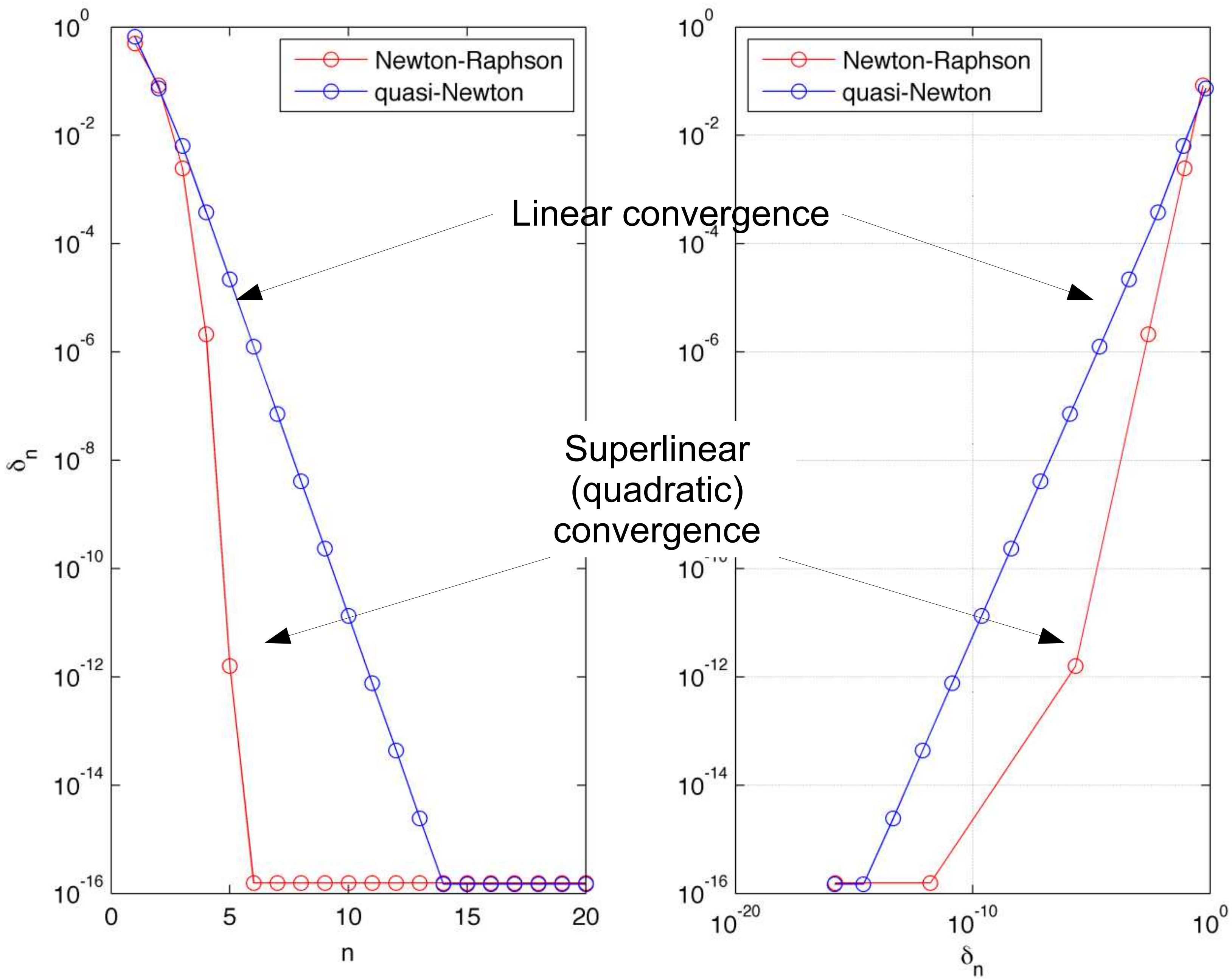
The number of accurate digits increases by a fixed amount at each iteration

Newton-Raphson :

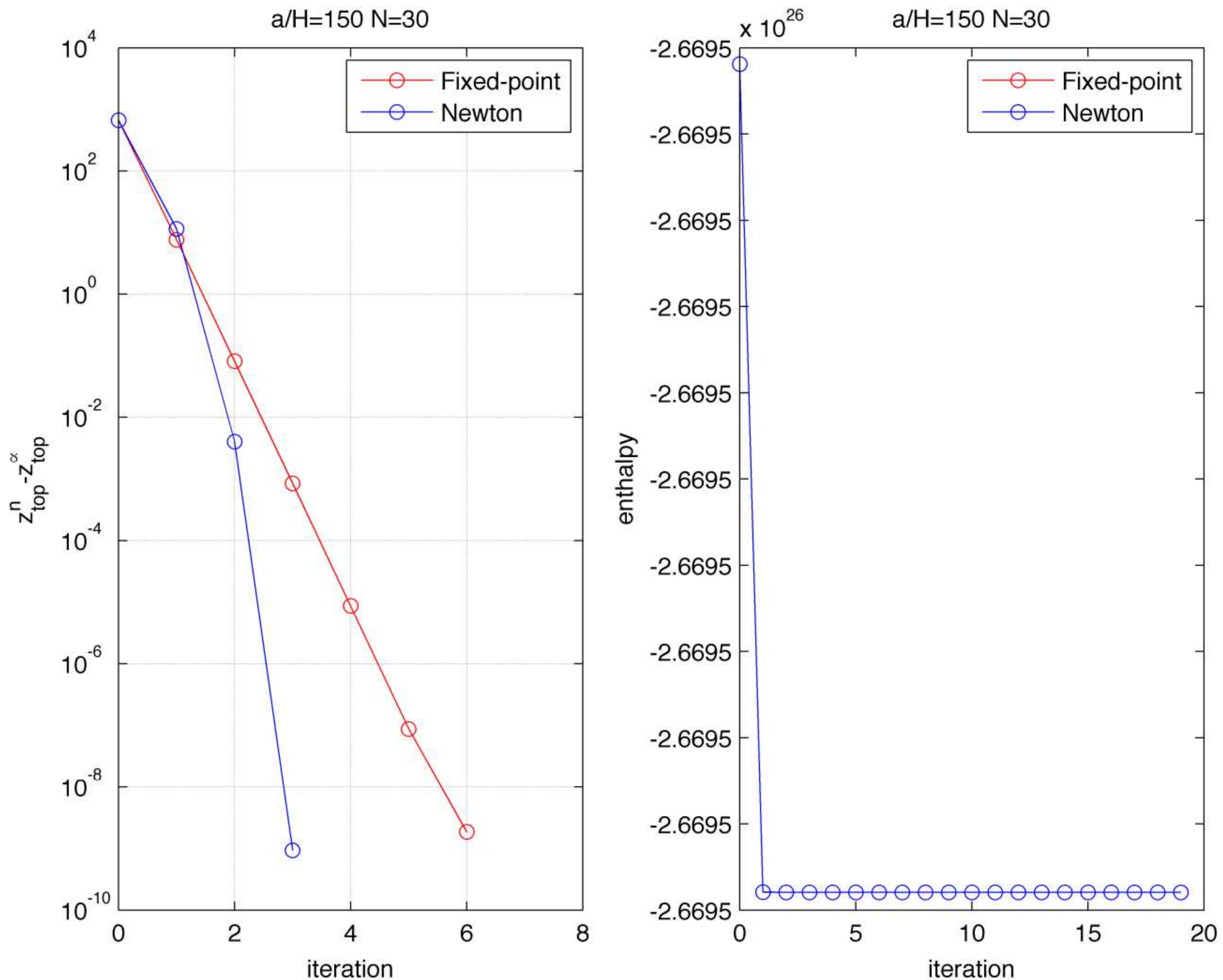
$$\delta_{n+1} = c.\delta_n^{\,2} \quad \Rightarrow \quad \log(c.\delta_n) = C.2^n$$

The number of accurate digits doubles at each iteration !

Fixed-point versus Newton-Raphson



Hydrostatic adjustment



Hydrostatic adjustment