A scenic sunset over a body of water. The sky is filled with large, dark clouds, with the sun breaking through in the upper right, creating a bright glow and rays of light. The sun's reflection is visible on the water's surface. In the foreground, a wooden pier extends from the bottom left towards the center. A small boat is visible on the water in the middle ground. The overall atmosphere is serene and dramatic.

## *Compléments algorithmiques:*

- *minimisation dans l'analyse variationnelle*
- *assimilation en temps réel et réanalyse*

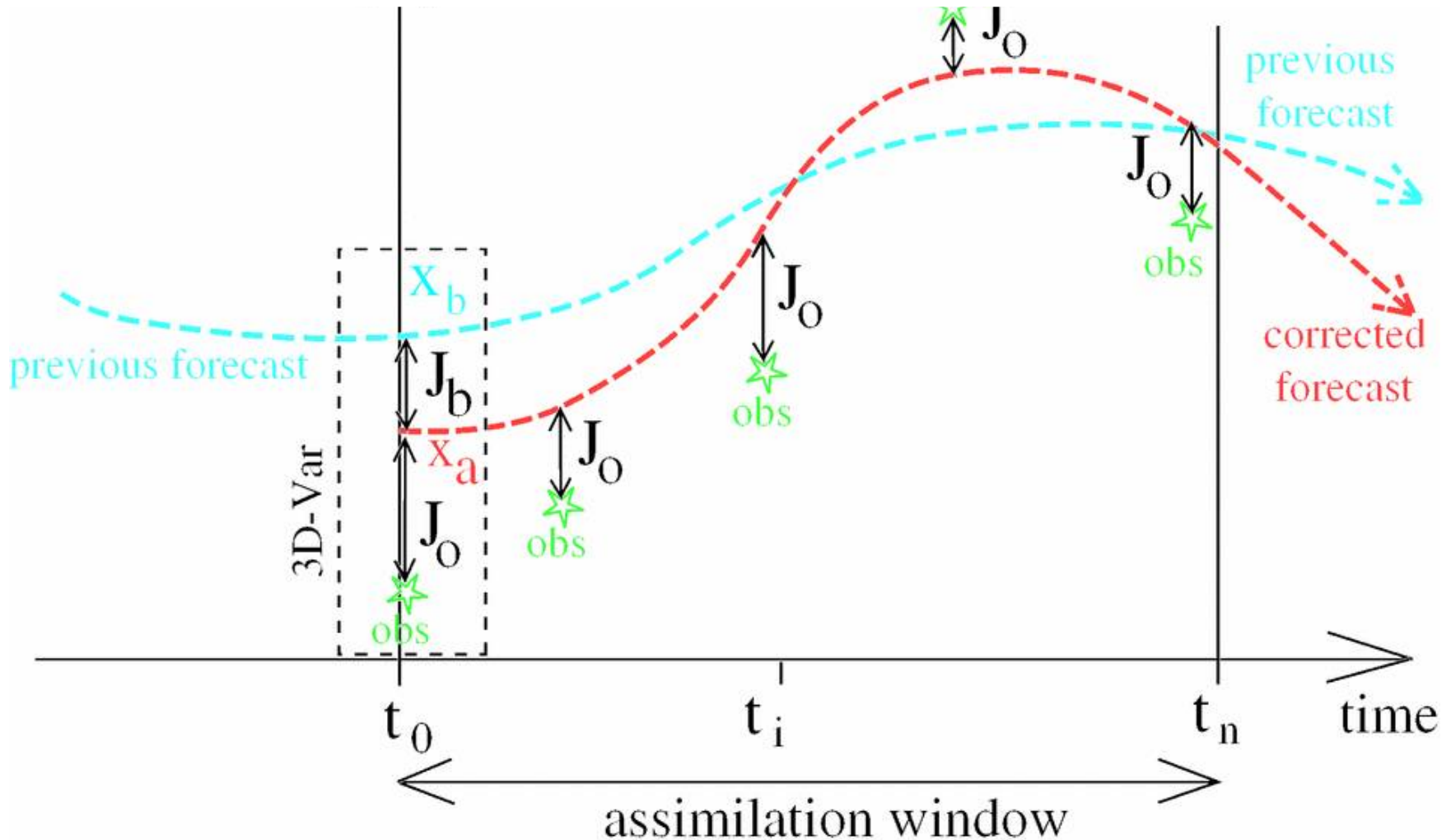
## **Minimisation en analyse variationnelle:**

- comment calculer le gradient de  $J$
- préconditionnement de  $J$
- le cas du 4D-Var

# Assimilation 4DVar: généralisation du BLUE à un historique d'observations, à l'aide d'un modèle de prévision M

$$J(x) = (x - x_b)^T B^{-1} (x - x_b) + \sum_i (y_i - H_i[\mathbf{M}_i(x)])^T R_i^{-1} (y_i - H_i[\mathbf{M}_i(x)])$$

La variable de minimisation est l'état  $x$  du modèle au début de la fenêtre.



# Minimisation du 4DVar

fonction-coût du 4DVar à minimiser:

$$x \rightarrow J(x) = (x - x_b)^T B^{-1} (x - x_b) + \sum_k (y_k - H_k \circ M_k[x])^T R_k^{-1} (y_k - H_k \circ M_k[x])$$

avec le modèle de prévision  $M_k$ :  $x=x(t_0) \rightarrow x(t_k)=M_k(x)$

le gradient devient  $2 B^{-1} (x - x_b) - 2 \sum_k M_k^T H_k^T R_k^{-1} (y_k - H_k \circ M_k[x])$

or  $M$  est une succession de pas de temps:  $M_k = N_k N_{k-1} N_{k-2} \dots N_1$

en factorisant on organise le calcul du gradient de droite ainsi:

- 1) pour chaque  $x_i$ , intégrer et stocker la '**trajectoire**' (  $M_1(x)$  ,  $M_2(x)$ ,...,  $M_k(x)$  )
- 2) calculer les écarts obs-trajectoire:  $d_k = ( y_k - H_k \circ M_k[x] )$
- 3) les multiplier par  $H_k^T R_k^{-1} d_k = e_k$
- 4) intégrer et stocker le '**modèle adjoint forcé**':  $z_k=0$ ,  $z_{k-1} = M_{k-1} z_k + e_k$

Chaque itération de la minimisation coûte 1 intégration du modèle de prévision + 1 intégration adjointe

Le modèle est parallélisable, mais pas la minimisation

# rappel mathématique: dérivée vectorielle, gradient

car la plupart des minimiseurs utilisent la dérivée de J

Soit F une application vectorielle  $\mathbb{R}^n \rightarrow \mathbb{R}^p$  :

- la dérivée (ou **différentielle**) de F est l'opérateur linéaire F' tel que

$$\forall(x, h) \quad F(x+h) = F(x) + F'(x) h + O(\|h\|^2)$$

- F'(x) est une matrice à p lignes et n colonnes
- les coefficients  $F_{ij}$  de la matrice F' sont les dérivées partielles  $\partial F_i / \partial x_j$  de F. Ils mesurent la sensibilité de la i-ième sortie de F par rapport à sa j-ième entrée, au voisinage de x.

*Cas de la fonction-coût J de l'analyse variationnelle:*

- J(x) est un scalaire, J' est un vecteur (identifié à une forme linéaire)  $\nabla J$  appelé **gradient de J**.

• la dérivée de  $\mathbf{J}(\mathbf{x}) = (\mathbf{x} - \mathbf{x}_b)^T \mathbf{B}^{-1} (\mathbf{x} - \mathbf{x}_b) + (\mathbf{y} - \mathbf{H}[\mathbf{x}])^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{H}[\mathbf{x}])$

est le vecteur  $\nabla \mathbf{J}(\mathbf{x}) = 2 \mathbf{B}^{-1} (\mathbf{x} - \mathbf{x}_b) - 2 \mathbf{H}^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{H}[\mathbf{x}])$

## calcul de la dérivée de J par code adjoint

• la dérivée de  $J(\mathbf{x}) = (\mathbf{x} - \mathbf{x}_b)^T \mathbf{B}^{-1} (\mathbf{x} - \mathbf{x}_b) + (\mathbf{y} - \mathbf{H}[\mathbf{x}])^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{H}[\mathbf{x}])$

est le *gradient*  $\nabla J(\mathbf{x}) = 2 \mathbf{B}^{-1} (\mathbf{x} - \mathbf{x}_b) - 2 \mathbf{H}'^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{H}[\mathbf{x}])$

Si  $\mathbf{x}$  ou  $\mathbf{y}$  sont grands, et  $\mathbf{H}$  est complexe, on évite de calculer la matrice .

$\mathbf{H}$  est une composée d'applications (une suite de lignes de code)

$\mathbf{H}_1 \circ \mathbf{H}_2 \circ \dots \circ \mathbf{H}_m$

$\mathbf{H}'$  est sa différentielle = produit des différentielles:

$\mathbf{H}'_1 \mathbf{H}'_2 \dots \mathbf{H}'_m$

$\mathbf{H}'^T$  est sa transposée = produit des transposées:

$\mathbf{H}'_m{}^T \dots \mathbf{H}'_2{}^T \mathbf{H}'_1{}^T$

c'est le **code adjoint de  $\mathbf{H}$**  = suite inverse des transposées des lignes de code (faisable par différentiation automatique)

# Codage d'un modèle adjoint

**Déf générale:** pour A opérateur linéaire, A\* tel que  $\langle Ax, y \rangle = \langle x, A^*y \rangle$

**Déf pour un code de modélisation:** transposée de la différentielle de l'opérateur correspondant à  $x_{\text{forecast}} = M(x_{\text{ini}})$

## Construction:

1) identifier l'état de linéarisation: *trajectoire* temporelle  $x_0 \ x_1 \ \dots \ x_T$

2) différencier  $M = M_T \circ M_{T-1} \circ \dots \circ M_1$  au voisinage de cette trajectoire:  $M' = M'_T \ M'_{T-1} \ \dots \ M'_1$  (produit des matrices des différentielles) *modèle linéaire tangent*

3) transposer:  $M^* = (M')^T = M'^T_1 \ M'^T_2 \ \dots \ M'^{TT} = M^*_1 \ M^*_2 \ \dots \ M^*_T$

modèle adjoint

# Codage d'un modèle adjoint

**Concrètement:** le modèle est un code séquentiel = succession d'opérateurs élémentaires

$$\langle dy_T, M' dx_0 \rangle = \langle M^* dy_T, x_0 \rangle$$

**Construction:**

1) mémoriser la *trajectoire*  $x_0, x_1, \dots, x_T = f(x_0)$

2) *modèle linéaire tangent*: séquence des opérateurs  $M' = M'_T, M'_{T-1}, \dots$   
 $M'_1 = f(x_0, dx_0)$

3) *modèle linéaire adjoint*  $M^* = M^*_T, M^*_{T-1}, \dots, M^*_1 = f(x_0, dy_T)$

En 4D-Var: calcul du gradient avec de  $M_k^T$

$$H_k^T R_k^{-1} (y_k - H_k \circ M_k[x])$$

vecteur colonne



# Codage d'un modèle adjoint - exemples

**code direct:**

$$x2 = a * x1 + b$$

$$x2 = \log(x1)$$

$$x3 = a * x1 + b * x2$$

début de scope de x1

fin de scope de x1

**code adjoint:**

$$dx1 = dx1 + a * dx2$$

$$dx1 = dx1 + dx2 / x1$$

$$dx1 = dx1 + a * dx3$$

$$dx2 = dx2 + b * dx3$$

fin de scope de x1

$$dx1 = 0$$

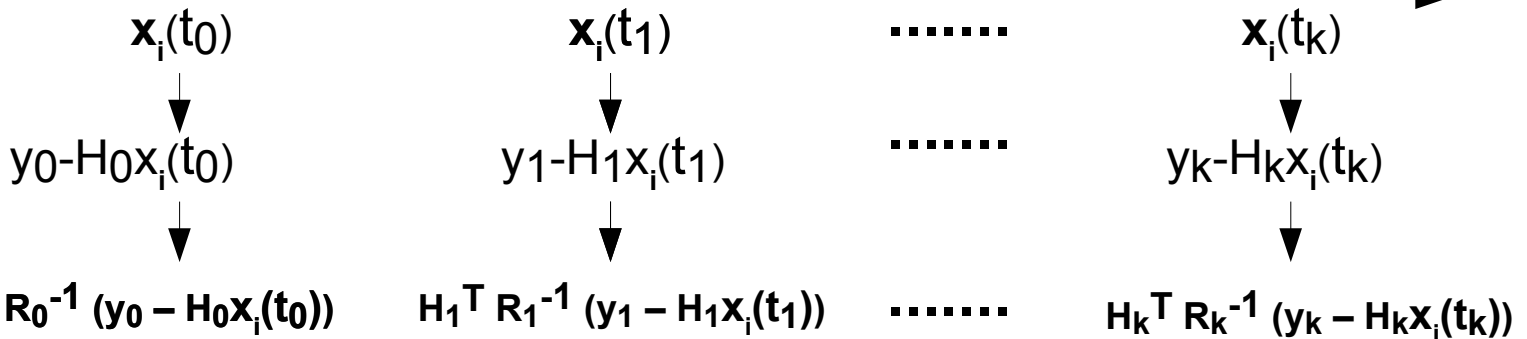
# Minimisation du 4DVar

fonction-coût du 4DVar à minimiser:

$$\mathbf{x} \rightarrow \mathbf{J}(\mathbf{x}) = (\mathbf{x} - \mathbf{x}_b)^T \mathbf{B}^{-1} (\mathbf{x} - \mathbf{x}_b) + \sum_{\mathbf{k}} (\mathbf{y}_k - \mathbf{H}_k \circ \mathbf{M}_k[\mathbf{x}])^T \mathbf{R}_k^{-1} (\mathbf{y}_k - \mathbf{H}_k \circ \mathbf{M}_k[\mathbf{x}])$$

gradient:  $\nabla \mathbf{J}(\mathbf{x}) = 2 \mathbf{B}^{-1} (\mathbf{x} - \mathbf{x}_b) - 2 \sum_{\mathbf{k}} \mathbf{M}_k^T \mathbf{H}_k^T \mathbf{R}_k^{-1} (\mathbf{y}_k - \mathbf{H}_k \circ \mathbf{M}_k[\mathbf{x}])$

$\mathbf{x}_i(t_0) \rightarrow$  prévision modèle M:  $\mathbf{M}(t_0 \rightarrow t_1)$      $\mathbf{M}(t_1 \rightarrow t_2)$     .....    temps  $\rightarrow$



prévision adjointe  $\mathbf{M}^T$ :  $\mathbf{M}^T(t_1 \rightarrow t_0)$      $\mathbf{M}^T(t_2 \rightarrow t_1)$     .....    temps  $\leftarrow$

$$\nabla \mathbf{J}(\mathbf{x}_i) = H_0^T R_0^{-1} (y_0 - H_0 \mathbf{x}_i(t_0)) + \mathbf{M}^T(t_1 \rightarrow t_0) \left[ H_1^T R_1^{-1} (y_1 - H_1 \mathbf{x}_i(t_1)) + \dots \left[ \dots + H_k^T R_k^{-1} (y_k - H_k \mathbf{x}_i(t_k)) \dots \right] \dots \right]$$

minimisation  $\mathbf{x}_{i+1} = F(\mathbf{x}_i, \nabla \mathbf{J}(\mathbf{x}_i))$

# préconditionnement de la minimisation

• on veut minimiser  $J(\mathbf{x}) = (\mathbf{x} - \mathbf{x}_b)^T \mathbf{B}^{-1} (\mathbf{x} - \mathbf{x}_b) + (\mathbf{y} - \mathbf{H}[\mathbf{x}])^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{H}[\mathbf{x}])$

la dérivée seconde est  $\mathbf{J}'' = 2\mathbf{B}^{-1} + 2\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}$

la vitesse de minimisation est améliorée si  $J$  est à peu près diagonale (minimiser le ratio entre +grande et +petite valeur propre)

Exemple: preconditionnement par changement de variable:

si  $\mathbf{B}^{-1} = \mathbf{L}^{-T} \mathbf{L}$  avec  $\mathbf{L}$  inversible: on pose  $\mathbf{z} = \mathbf{L}(\mathbf{x} - \mathbf{x}_b)$

alors  $J(\mathbf{x}) = G(\mathbf{z}) = \mathbf{z}^T \mathbf{z} + (\mathbf{y} - \mathbf{H}[\mathbf{x}])^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{H}[\mathbf{x}])$

et  $G'' = \mathbf{I} + \dots$

si le nombre d'obs est  $\ll \dim \mathbf{x}$ , cela permet de gagner plusieurs ordres de grandeur sur la vitesse de minimisation

## **Assimilation en temps réel:**

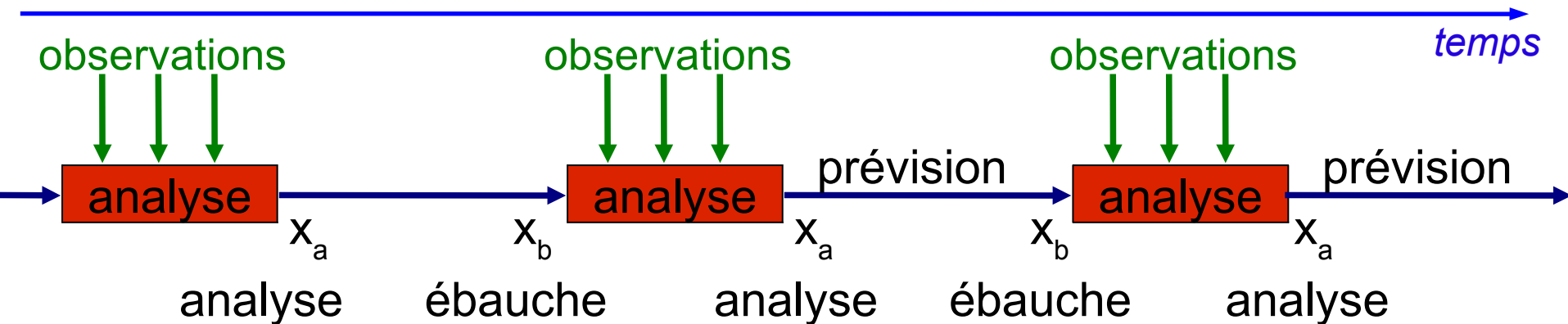
- notion de cutoff des observations
- production de prévisions en temps réel
- réanalyse

# Assimilations de données temps réel

- :
- pour faire des prévisions
  - pour le suivi climatique

Problèmes:

- les observations arrivent plus ou moins tard
- des défaillances peuvent rompre la chaîne d'assimilation



# Temps réel: notion de 'cutoff'

Plus on attend, et...

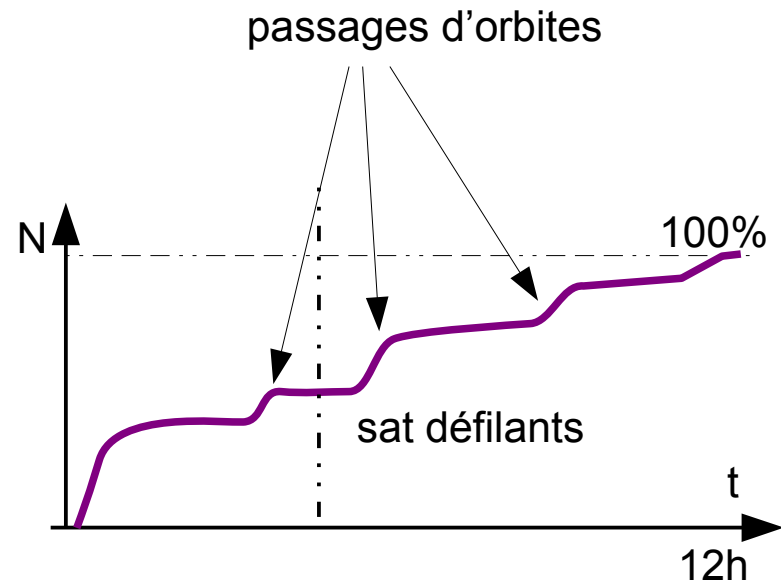
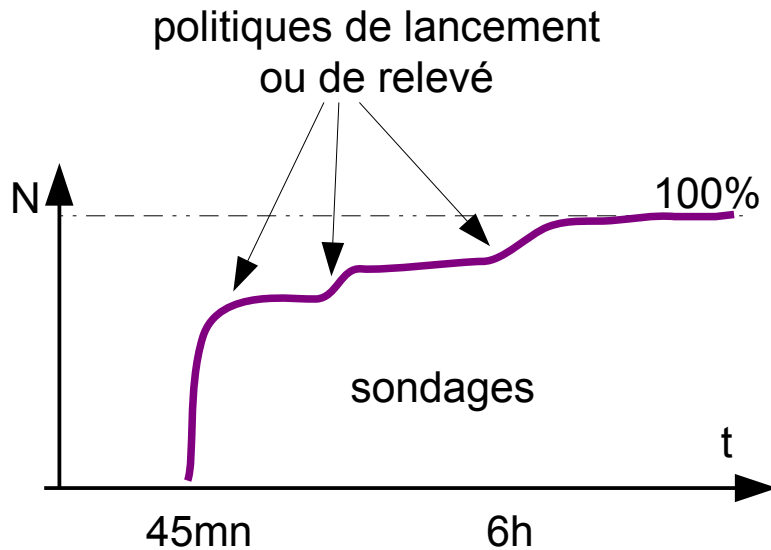
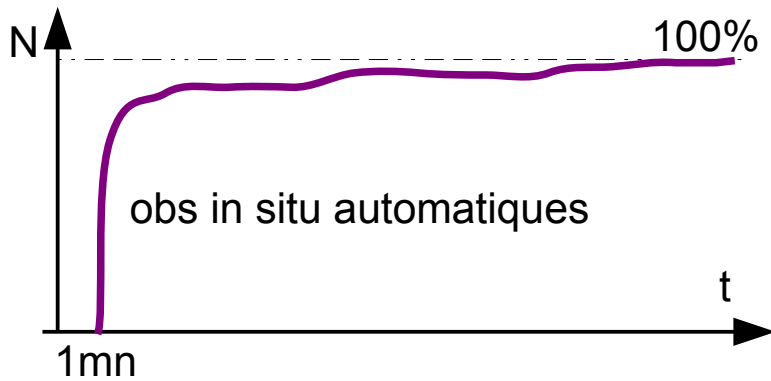
- ...plus il y a d'obs à assimiler -> meilleures analyses -> meilleures ébauches
- ...moins l'analyse est utile, parce que les prévisions seront disponibles plus/trop tard pour les usagers.

Chaque type d'obs présente des **latences**: délai entre mesure physique, et disponibilité pour le système d'analyse:

- réseau de surface in situ automatisé: <1mn
- sondages: 45mn (temps d'ascendance)
- radar: 5 à 15mn (période de balayage)
- satellite géostationnaire: quelques mn (segment sol) + temps de balayage de l'imageur
- satellite défilant: de quelques mn à plusieurs heures en fonction de l'orbitographie et des stations de réception au sol

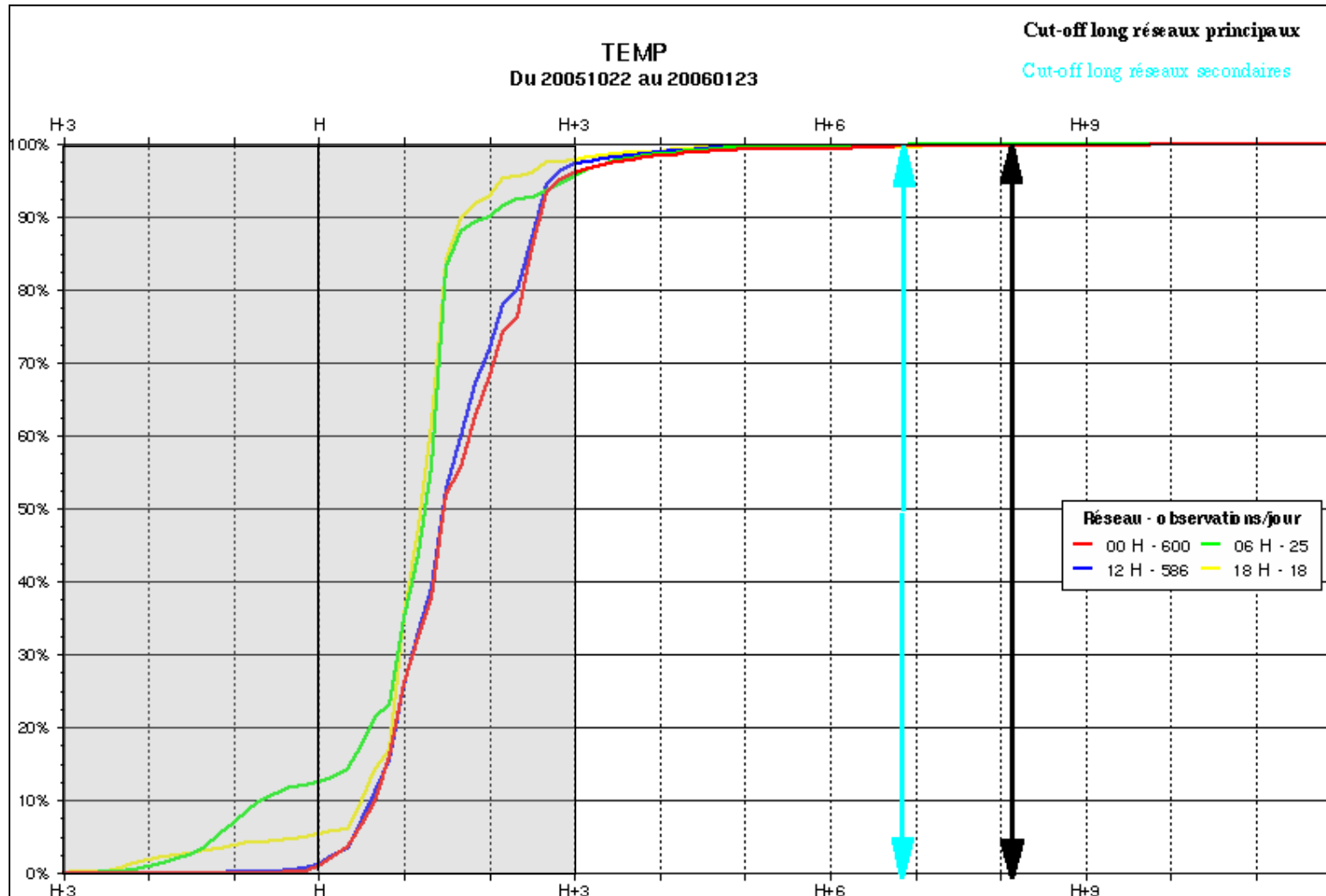
# Temps réel: notion de 'cutoff' (relevé)

diagrammes typiques:  $N_{obs} = f(t_{latence})$



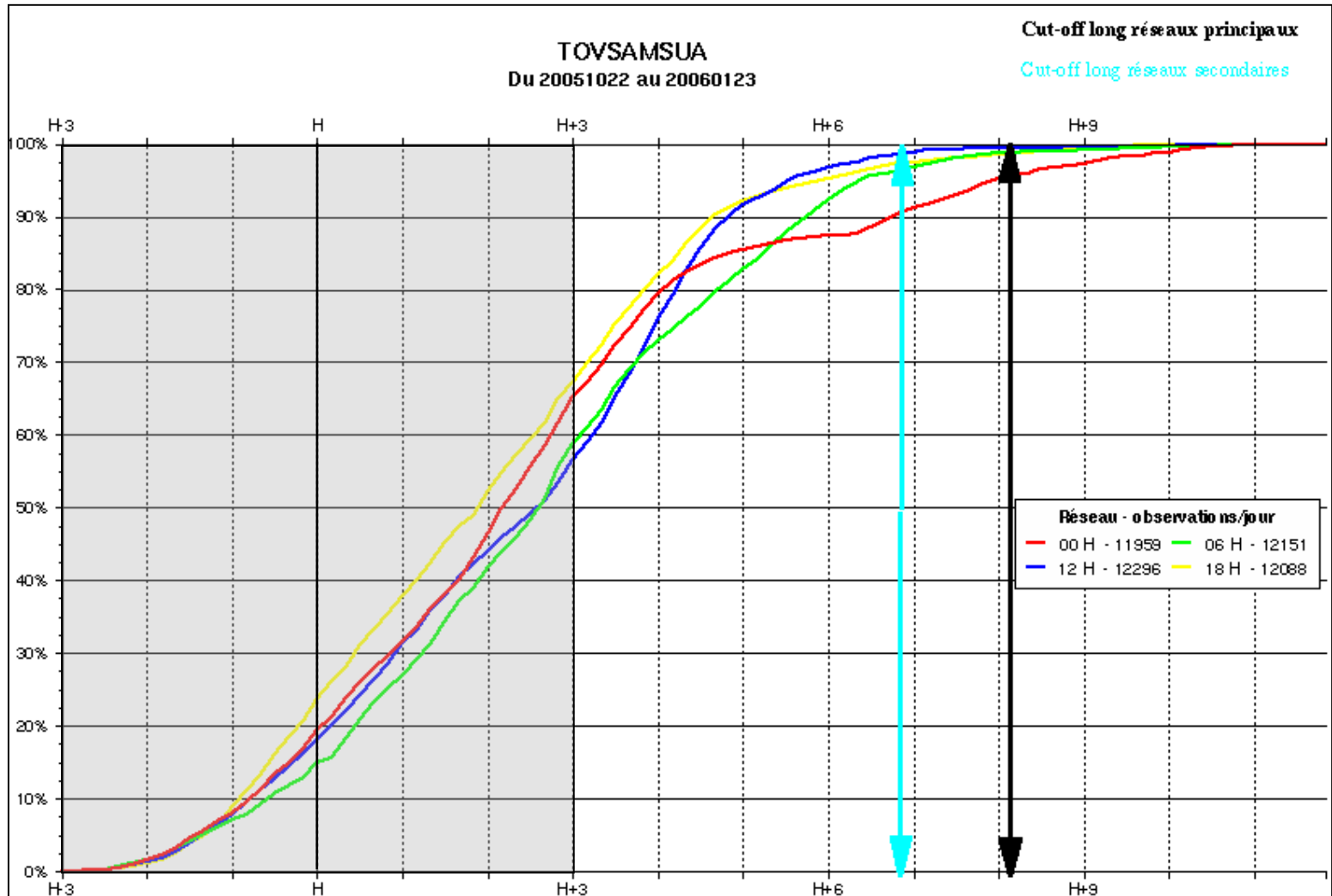
'**cutoff**': instant à partir duquel on cesse d'attendre les observations, et on commence à calculer l'analyse.

# Disponibilité des radiosondages d'une heure donnée en fonction du cut-off





# Disponibilité des radiances sat d'une heure donnée en fonction du cut-off



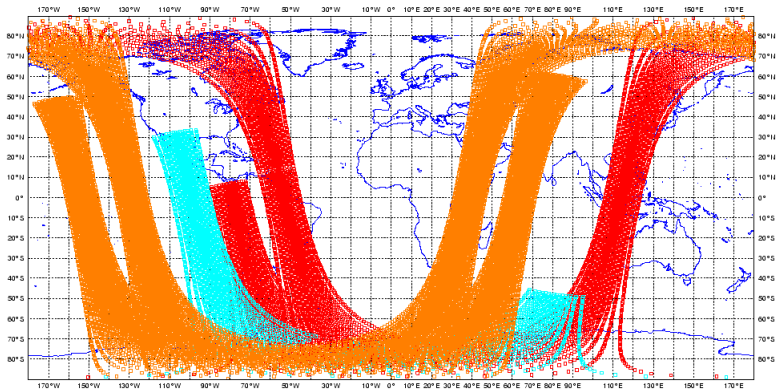
# Distribution spatiale des obs sat en fonction de l'heure d'analyse et du cutoff

METEO-FRANCE couverture de donnees - ATOVS AMSU-A

2006/01/16 00H UTC cut-off court

Nombre total d'observations avant screening : 29790

7125 NOAA-15    2535 NOAA-16    0 NOAA-17    20130 AQUA

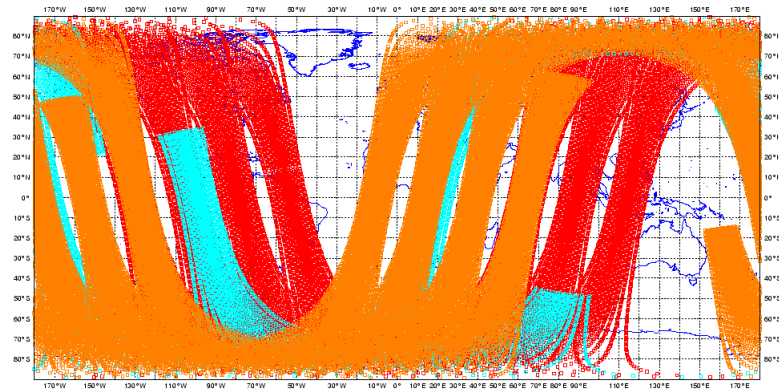


METEO-FRANCE couverture de donnees - ATOVS AMSU-A

2006/01/16 00H UTC cut-off long

Nombre total d'observations avant screening : 68813

20385 NOAA-15    8745 NOAA-16    0 NOAA-17    39683 AQUA



ana 0hTU, 30min de cutoff

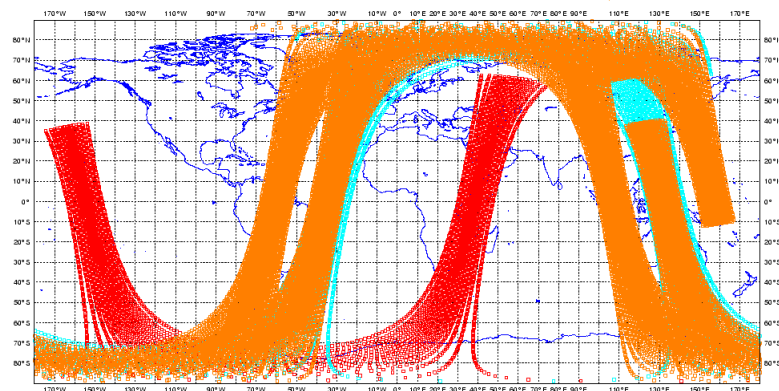
ana 0hTU, 5h de cutoff

ana 0hTU, 30min de cutoff

2006/01/16 06H UTC cut-off court

Nombre total d'observations avant screening : 35640

4440 NOAA-15    6345 NOAA-16    0 NOAA-17    24855 AQUA



121h-03hJ

103h-09hJ

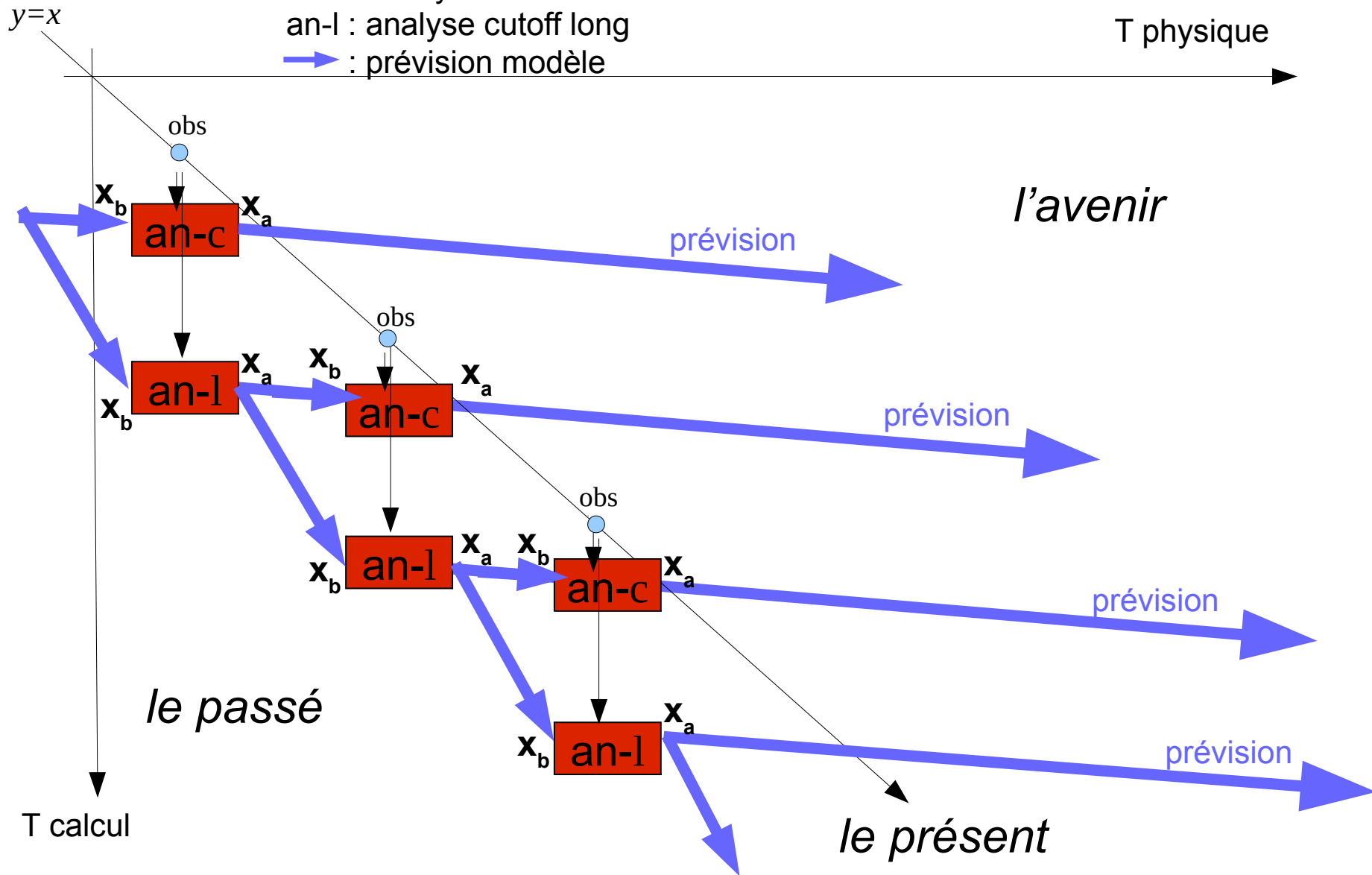
# Centre de prévision temps réel

## Organisation temporelle des opérations

an-c : analyse cutoff court

an-l : analyse cutoff long

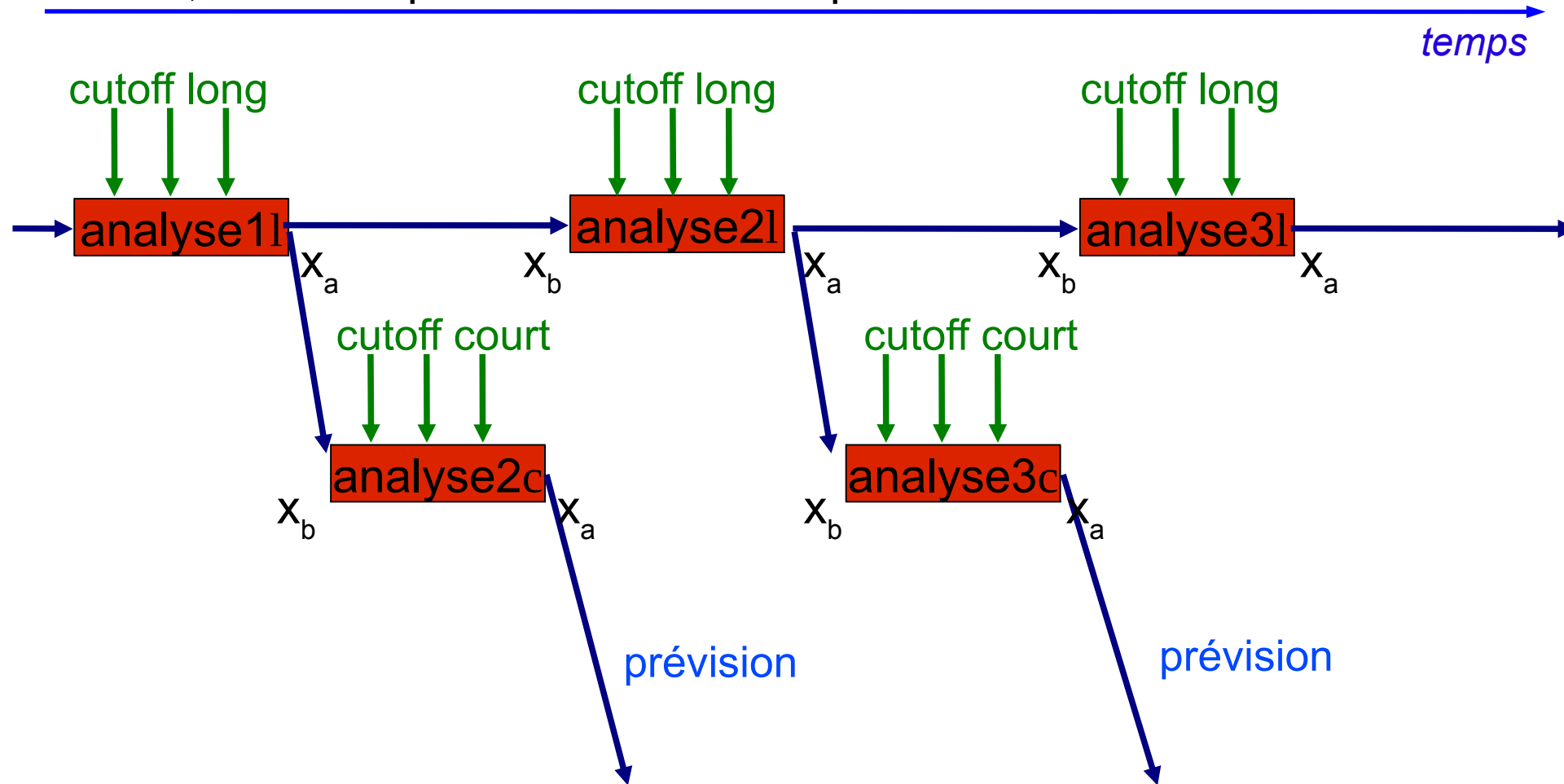
→ : prévision modèle



# Centre de prévision temps réel

On fait tourner 2 chaînes couplées:

- une à cutoff long (~5h), de haute qualité mais tardive
- une à cutoff court (~10mn), non cyclée, pour la production à flux tendus, avec uniquement les obs les plus fraîches



## En résumé sur le cutoff

On calcule chaque analyse plusieurs fois:

- une fois très tôt (cutoff court), pour produire analyse et prévision à **flux tendu**.
- une fois plus tard (cutoff long), pour produire la **meilleure ébauche possible** pour la prochaine analyse à flux tendu.
- seules les analyses à cutoff long sont cyclées dans l'assimilation

## Réanalyses

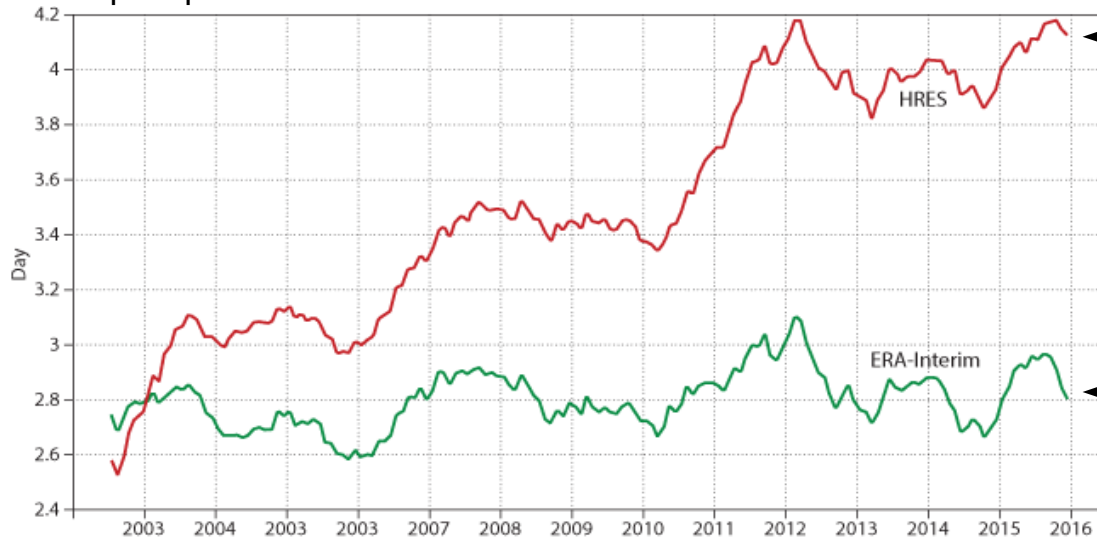
=**analyses à cutoff très long** (pour suivi climatique, études de cas...). Avantages:

- utilise **obs non opérationnelles** (réseaux/satellites de recherche, obs historiques...)
- **contrôle qualité** et **correction de biais** optimisés
- algorithmes modernes de modèle et d'assimilation

# Applications des réanalyses

Evolution de la qualité des prévisions, indépendamment des évolutions du réseau d'observation:

score de précipitation

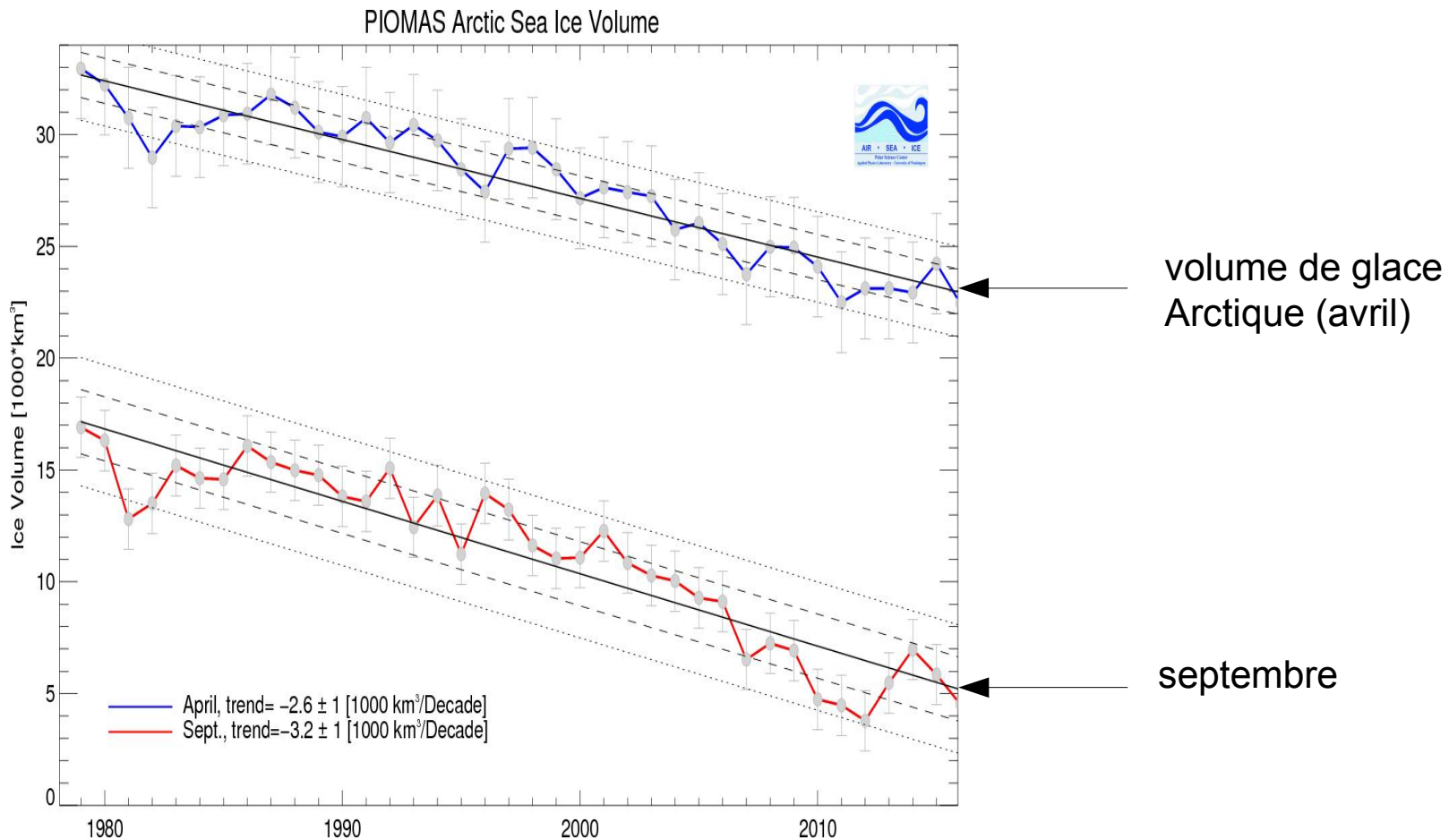


modèle CEPMMT  
au jour le jour

réanalyse sur 14 ans, à  
modèle constant

# Applications des réanalyses

Evolution climatique corrigée des évolutions des observations et des modèles, avec barres d'erreur:



Schweiger, A., R. Lindsay, J. Zhang, M. Steele, H. Stern,  
*Uncertainty in modeled arctic sea ice volume, J. Geophys. Res.,*  
doi:10.1029/2011JC007084, 2011

*Fin*