Réseaux de neurones parcimonieux à grande diversité d'apprentissage

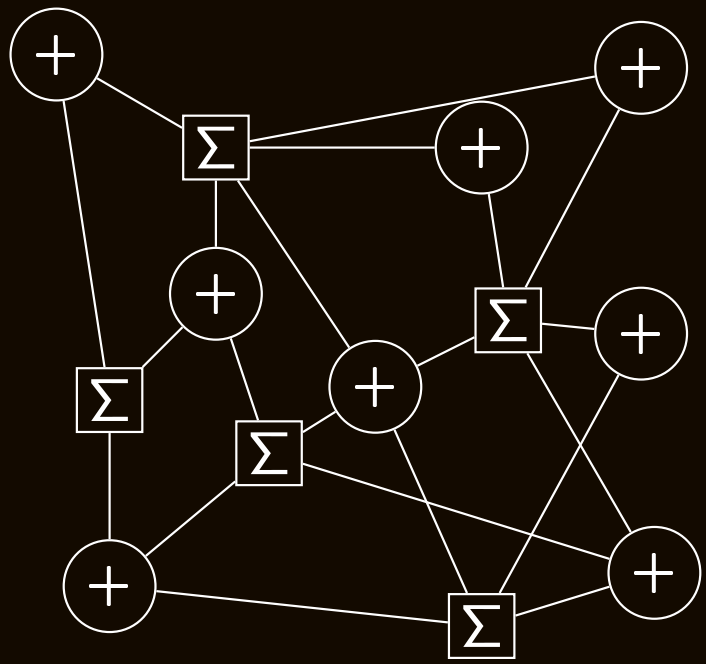
Vincent Gripon Claude Berrou

Télécom Bretagne, Lab-STICC

16 décembre 2010

Contexte : croisement entre théorie de l'information et computation neurale

Décodeur de code LDPC



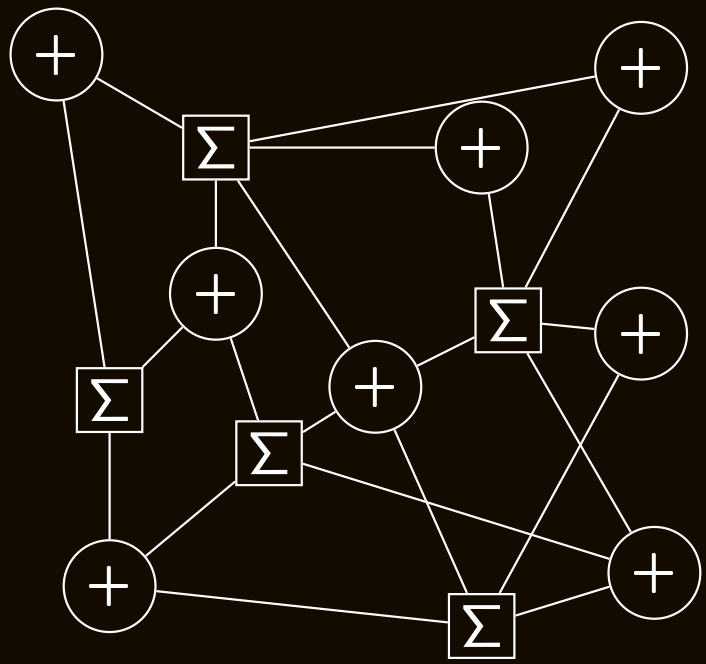
Décodeur néocortical

Forte analogie

Mais où sont passées les relations de parité ?

Contexte : croisement entre théorie de l'information et computation neurale

Décodeur de code LDPC



Décodeur néocortical

Forte analogie

Mais où sont passées les relations de parité ?

Contexte : Analogies entre le décodage correcteur et le décodage neural

Analogies

Décodage correcteur		Décodage neural
Point fixe du décodage	$\leftrightarrow$	Souvenir unique et non confus
Distance minimale	$\leftrightarrow$	Souvenirs séparables
Énorme diversité de combinaisons	$\leftrightarrow$	Grande capacité de mémorisation
Densité très faible des graphes	$\leftrightarrow$	Densité faible du réseau neural

Résilience, homéostasie, synchronisation, bruit...

Dissemblances

Girth maximal	$\leftrightarrow$	Girth quelconque
Deux types de processeurs	$\leftrightarrow$	Un seul type de “processeur”
Messages choisis	$\leftrightarrow$	Messages subis
Messages linéairement liés	$\leftrightarrow$	Messages quelconques

Contexte : Analogies entre le décodage correcteur et le décodage neural

Analogies

Décodage correcteur		Décodage neural
Point fixe du décodage	$\leftrightarrow$	Souvenir unique et non confus
Distance minimale	$\leftrightarrow$	Souvenirs séparables
Énorme diversité de combinaisons	$\leftrightarrow$	Grande capacité de mémorisation
Densité très faible des graphes	$\leftrightarrow$	Densité faible du réseau neural

Résilience, homéostasie, synchronisation, bruit...

Dissemblances

Girth maximal	$\leftrightarrow$	Girth quelconque
Deux types de processeurs	$\leftrightarrow$	Un seul type de “processeur”
Messages choisis	$\leftrightarrow$	Messages subis
Messages linéairement liés	$\leftrightarrow$	Messages quelconques

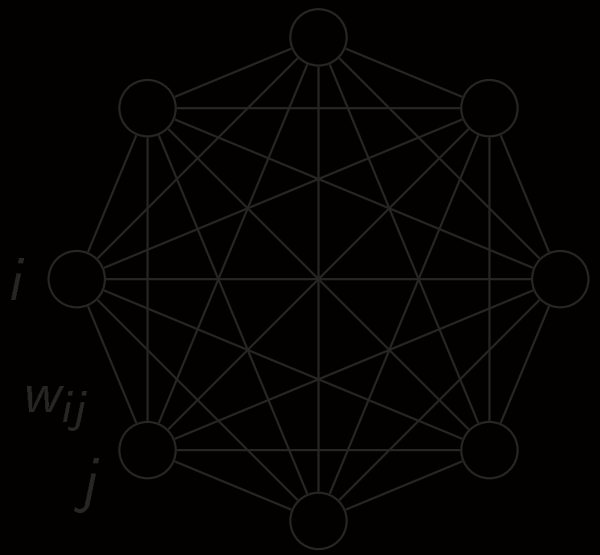
Mémoires associatives, état de l'art

Principe

Deux opérations :

- Apprendre un message,
- Remémorer un message précédemment appris en présence d'erreurs et/ou d'effacements.

Exemple avec les réseaux de Hopfield



- Apprentissage : M messages $\mathbf{d}^m$ binaires : $w_{ij} = \sum_{m=1, i \neq j}^{M} d_i^m d_j^m$,
- Remémoration : Répéter $\forall i, v_i \leftarrow \mathrm{sgn}(\sum_{j \neq i} v_j w_{ij})$.

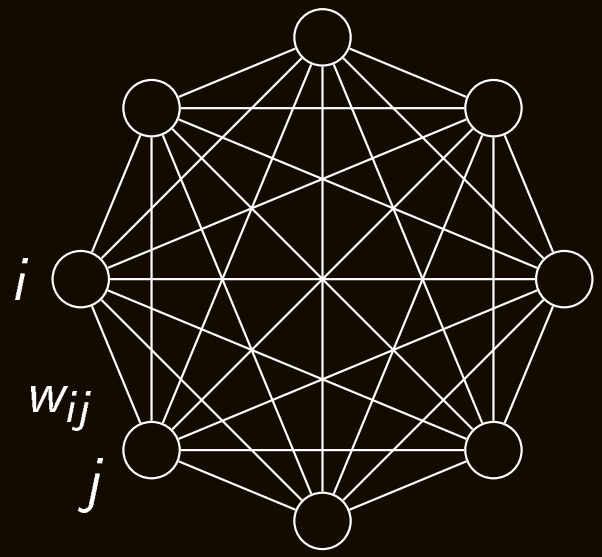
Mémoires associatives, état de l'art

Principe

Deux opérations :

- Apprendre un message,
- Remémorer un message précédemment appris en présence d'erreurs et/ou d'effacements.

Exemple avec les réseaux de Hopfield



- Apprentissage : M messages $\mathbf{d}^m$ binaires : $w_{ij} = \sum_{m=1, i \neq j}^{M} d_i^m d_j^m$,
- Remémoration : Répéter $\forall i, v_i \leftarrow \mathrm{sgn}(\sum_{j \neq i} v_j w_{ij})$.

Performances et limites

Réseau de Hopfield

- Diversité : $\frac{n}{log(n)}$,
- Capacité : $\frac{n^2}{log(n)}$,
- Informations binaires stockées : $\frac{n(n-1)}{2} log_2(M+1)$,
- $\Rightarrow$ Efficacité $\approx \frac{2}{log(n) log_2(M+1)}$.
- Connexions sensibles, valeurs négatives, diversité et taille des messages $= f$(taille du réseau), messages à inversion près...

Limites théoriques pour un graphe complet sans boucles

- Capacité si connexions sur P niveaux : $\approx \frac{n^2}{2} log_2(P)$,
- Si efficacité 1 : $\approx \frac{n}{2} log_2(P)$ messages de longueur n,
- Si longueur k : $\approx \frac{n^2 log_2(P)}{2k}$.

Performances et limites

Réseau de Hopfield

- Diversité : $\frac{n}{log(n)}$,
- Capacité : $\frac{n^2}{log(n)}$,
- Informations binaires stockées : $\frac{n(n-1)}{2} log_2(M+1)$,
- $\Rightarrow$ Efficacité $\approx \frac{2}{log(n) log_2(M+1)}$.
- Connexions sensibles, valeurs négatives, diversité et taille des messages $= f$(taille du réseau), messages à inversion près...

Limites théoriques pour un graphe complet sans boucles

- Capacité si connexions sur P niveaux : $\approx \frac{n^2}{2} log_2(P)$,
- Si efficacité 1 : $\approx \frac{n}{2} log_2(P)$ messages de longueur n,
- Si longueur k : $\approx \frac{n^2 log_2(P)}{2k}$.

Performances et limites

Réseau de Hopfield

- Diversité : $\frac{n}{log(n)}$,
- Capacité : $\frac{n^2}{log(n)}$,
- Informations binaires stockées : $\frac{n(n-1)}{2} log_2(M+1)$,
- $\Rightarrow$ Efficacité $\approx \frac{2}{log(n) log_2(M+1)}$.
- Connexions sensibles, valeurs négatives, diversité et taille des messages $= f$(taille du réseau), messages à inversion près...

Limites théoriques pour un graphe complet sans boucles

- Capacité si connexions sur P niveaux : $\approx \frac{n^2}{2} log_2(P)$,
- Si efficacité 1 : $\approx \frac{n}{2} log_2(P)$ messages de longueur n,
- Si longueur k : $\approx \frac{n^2 log_2(P)}{2k}$.

Codage correcteur, approche classique

Codage systématique

- Codage d'un message de k bits : 01100..10010,
- Ajout d'une redondance de $n-k$ bits : 1110..011,
- La redondance est fonction du message,
- Le mot de code est obtenu par concaténation du message et de la redondance : 01100..100101110..011.

Décodage correcteur

- Un mot de code bruité est reçu,
- Le mot de code connu le plus proche est choisi,
- Plus les mots de codes sont distants, plus grandes sont les chances de retrouver le bon,
- $\Rightarrow$ Distance minimale d_{min}.

Codage correcteur, approche classique

Codage systématique

- Codage d'un message de k bits : 01100..10010,
- Ajout d'une redondance de $n - k$ bits : 1110..011,
- La redondance est fonction du message,
- Le mot de code est obtenu par concaténation du message et de la redondance : 01100..100101110..011.

Décodage correcteur

- Un mot de code bruité est reçu,
- Le mot de code connu le plus proche est choisi,
- Plus les mots de codes sont distants, plus grandes sont les chances de retrouver le bon,
- $\Rightarrow$ Distance minimale d_{min}.

Codage correcteur, approche classique

Codage systématique

- Codage d'un message de k bits : 01100..10010,
- Ajout d'une redondance de $n - k$ bits : 1110..011,
- La redondance est fonction du message,
- Le mot de code est obtenu par concaténation du message et de la redondance : 01100..100101110..011.

Décodage correcteur

- Un mot de code bruité est reçu,
- Le mot de code connu le plus proche est choisi,
- Plus les mots de codes sont distants, plus grandes sont les chances de retrouver le bon,
- $\Rightarrow$ Distance minimale d_{min}.

Codage correcteur, approche générale

Définition

- Un code est défini par l'ensemble de ses mots de code,
- Plus de partie systématique, association message↔mot de code quelconque.

Exemple : cliques

- Distance minimale atteinte quand un noeud diffère : $2(n-1) \approx 2n$,
- Rendement de code $\frac{n}{2}\frac{2}{n(n-1)} \approx \frac{1}{n} \Rightarrow$ Facteur de mérite environ 2.

Exemples

- Les mots de code sont des mots contenant exactement ω 1 et ne partageant pas plus de α 1 en commun deux à deux,
- *Constant weight code* de paramètre ω et α-recouvrant sur des mots de taille n : $C^n(\omega, \alpha)$.

Codage correcteur, approche générale

Définition

- Un code est défini par l'ensemble de ses mots de code,
- Plus de partie systématique, association message$\leftrightarrow$mot de code quelconque.

Exemple : cliques

- Distance minimale atteinte quand un noeud diffère : $2(n-1) \approx 2n$,
- Rendement de code $\frac{n}{2}\frac{2}{n(n-1)} \approx \frac{1}{n} \Rightarrow$ Facteur de mérite environ 2.

Exemples

- Les mots de code sont des mots contenant exactement ω 1 et ne partageant pas plus de α 1 en commun deux à deux,
- *Constant weight code* de paramètre ω et α-recouvrant sur des mots de taille n : $C^n(\omega, \alpha)$.

Codage correcteur, approche générale

Définition

- Un code est défini par l'ensemble de ses mots de code,
- Plus de partie systématique, association message$\leftrightarrow$mot de code quelconque.

Exemple : cliques

- Distance minimale atteinte quand un noeud diffère : $2(n-1) \approx 2n$,
- Rendement de code $\frac{n}{2}\frac{2}{n(n-1)} \approx \frac{1}{n} \Rightarrow$ Facteur de mérite environ 2.

Exemples

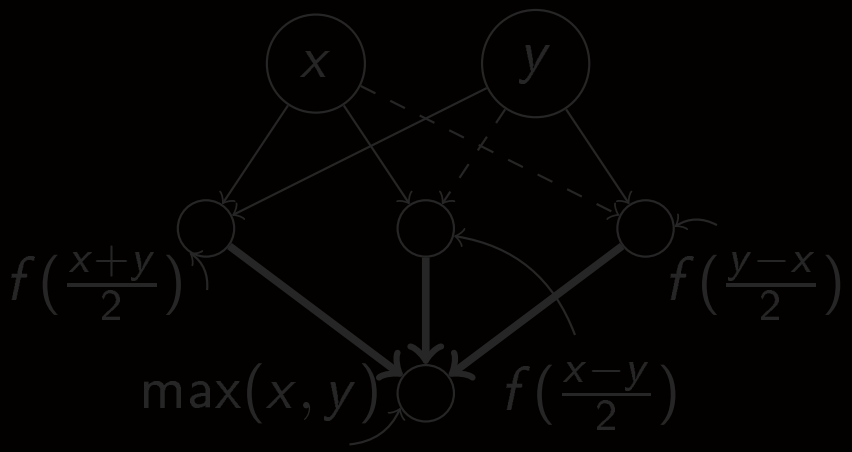
- Les mots de code sont des mots contenant exactement ω 1 et ne partageant pas plus de α 1 en commun deux à deux,
- *Constant weight code* de paramètre ω et α-recouvrant sur des mots de taille n : $C^n(\omega, \alpha)$.

Constant weight code et décodage neural

Code parcimonieux maximal

- $C^n(1,0)$ est l'ensemble des mots ne contenant qu'un seul 1 ($C^3(1,0) = \{100, 010, 001\}$),
- Très faible $d_{\min}$: 2 mais facilement décodable, faible en énergie,
- associable à la façon des *LDPC*...

Décodage neural



$\Rightarrow$

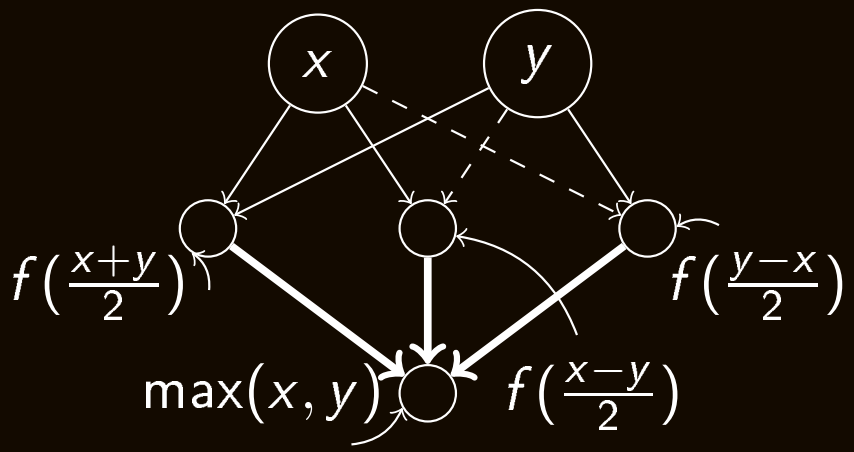
$$f(i) = \max(0, i)$$

Constant weight code et décodage neural

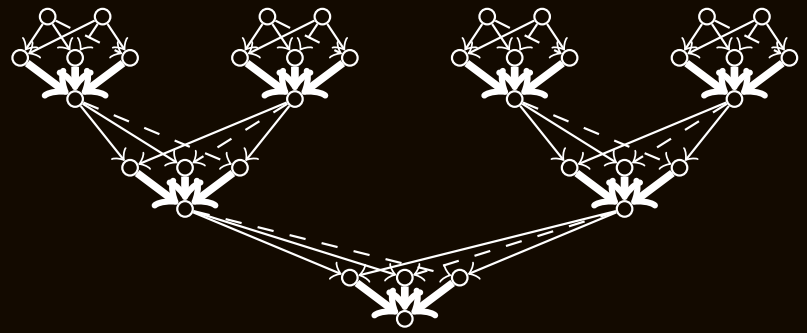
Code parcimonieux maximal

- $C^n(1,0)$ est l'ensemble des mots ne contenant qu'un seul 1 ($C^3(1,0) = \{100, 010, 001\}$),
- Très faible $d_{\min}$: 2 mais facilement décodable, faible en énergie,
- associable à la façon des *LDPC*...

Décodage neural



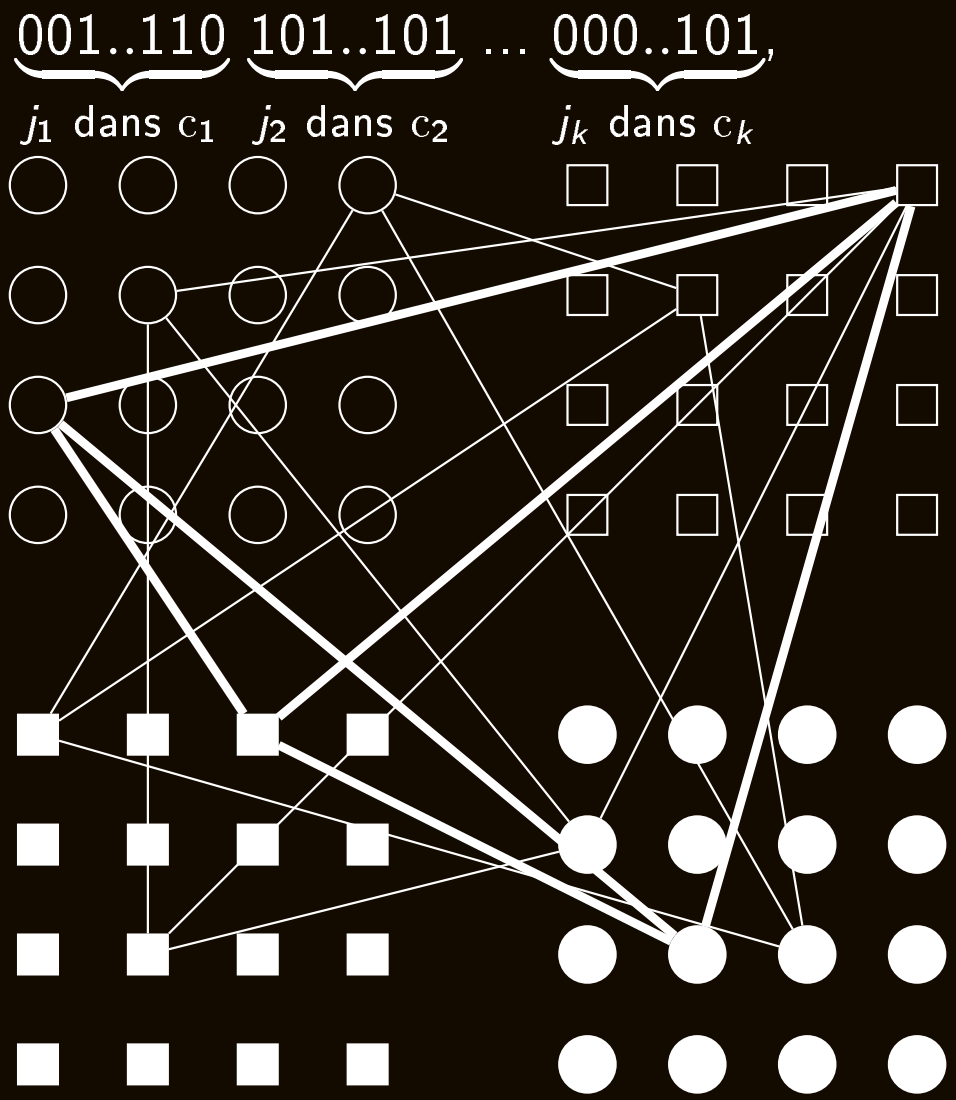
$\Rightarrow$



$$f(i) = \max(0, i)$$

Réseau de neurones à codage parcimonieux

Idée



- n neurones (fanaux),
- c clusters (ou blocs),
- κ bits pour adresser un cluster,
- $l = \frac{n}{c} = 2^{\kappa}$ neurones par cluster,
- $k = c\kappa$ bits par message appris,
- Parcimonie : un seul fanal actif par cluster.

Équations et limites

Apprentissage

- Valeur du fanal : $\mu_{bj}^{m} = 1$ si le neurone j du cluster b est associé au message m,
- $W_{b_1 j_1 b_2 j_2} = \min(\sum_{m=1, b_1 \neq b_2}^{M} \mu_{b_1 j_1}^{m} \mu_{b_2 j_2}^{m}, 1)$

Densité

- Après M messages aléatoires : $d \approx 1 - \left(1 - \frac{1}{l^2}\right)^{M}$,
- Une densité proche de 1 équivaut à l'incapacité de remémoration.

Limites

$$M_{\max} = \frac{(c-1)n^2}{2c^2 log_2(\frac{n}{c})}$$

Équations et limites

Apprentissage

- Valeur du fanal : $\mu_{bj}^{m} = 1$ si le neurone j du cluster b est associé au message m,
- $W_{b_1 j_1 b_2 j_2} = \min(\sum_{m=1, b_1 \neq b_2}^{M} \mu_{b_1 j_1}^{m} \mu_{b_2 j_2}^{m}, 1)$

Densité

- Après M messages aléatoires : $d \approx 1 - \left(1 - \frac{1}{l^2}\right)^{M}$,
- Une densité proche de 1 équivaut à l'incapacité de remémoration.

Limites

$$M_{\max} = \frac{(c-1)n^2}{2c^2 log_2(\frac{n}{c})}$$

Équations et limites

Apprentissage

- Valeur du fanal : $\mu_{bj}^{m} = 1$ si le neurone j du cluster b est associé au message m,
- $W_{b_1 j_1 b_2 j_2} = \min(\sum_{m=1, b_1 \neq b_2}^{M} \mu_{b_1 j_1}^{m} \mu_{b_2 j_2}^{m}, 1)$

Densité

- Après M messages aléatoires : $d \approx 1 - \left(1 - \frac{1}{l^2}\right)^M$,
- Une densité proche de 1 équivaut à l'incapacité de remémoration.

Limites

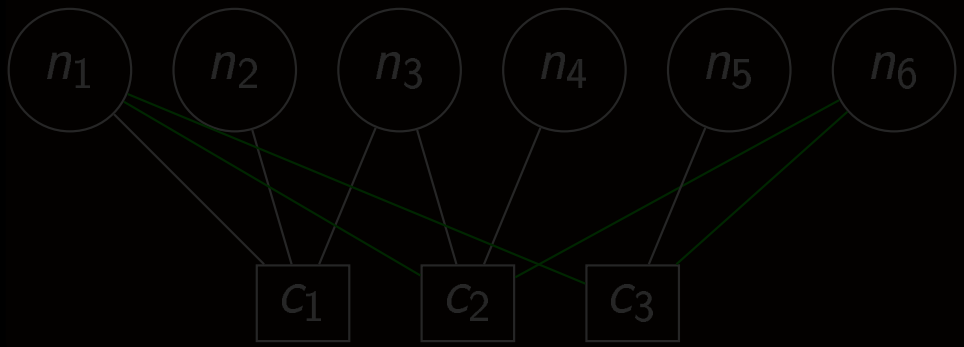
$$M_{\max} = \frac{(c-1)n^2}{2c^2 log_2(\frac{n}{c})}$$

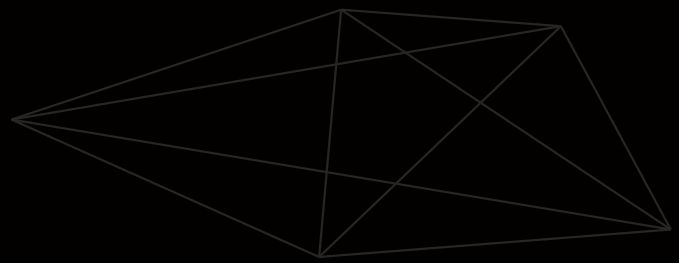
Remémoration

Processus itératif

- Globalement, neurones sommateurs :
  $\forall b_1, j_1 \, v_{b_1 j_1} \leftarrow \sum_{j_2, b_2 \neq b_1} W_{b_1 j_1 b_2 j_2} \mu_{b_2 j_2} + \gamma \mu_{b_1 j_1}$,
- Localement, *winner takes all* :
  - $\forall b, S^b_{\max} = \max_j v_{bj}$,
  - $\forall b, j, \mu_{bj} \leftarrow \begin{cases} 1 & \text{if } v_{bj} = S^b_{\max} \text{ and } S^b_{\max} \geq \sigma \\ 0 & \text{otherwise} \end{cases}$.

Corrélation *or not* corrélation



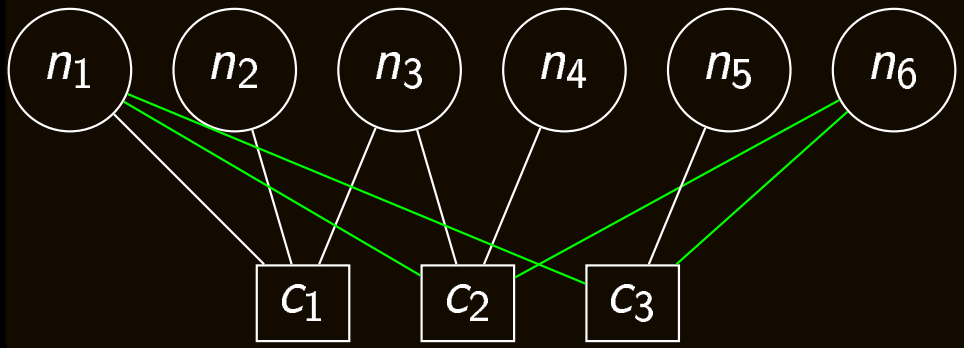


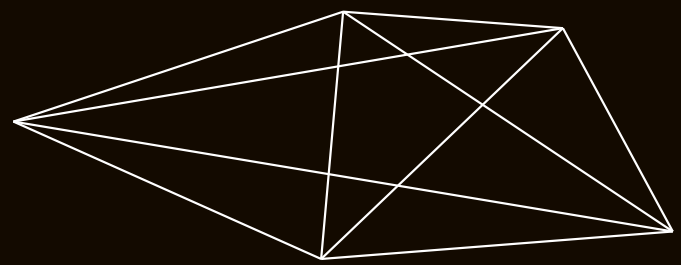
Remémoration

Processus itératif

- Globalement, neurones sommateurs :
  $$\forall b_1, j_1 \, v_{b_1 j_1} \leftarrow \sum_{j_2, b_2 \neq b_1} W_{b_1 j_1 b_2 j_2} \mu_{b_2 j_2} + \gamma \mu_{b_1 j_1},$$
- Localement, *winner takes all* :
  - $\forall b, S^b_{\max} = \max_j v_{bj}$,
  - $$\forall b, j, \mu_{bj} \leftarrow \begin{cases} 1 & \text{if } v_{bj} = S^b_{\max} \text{ and } S^b_{\max} \geq \sigma \\ 0 & \text{otherwise} \end{cases}.$$

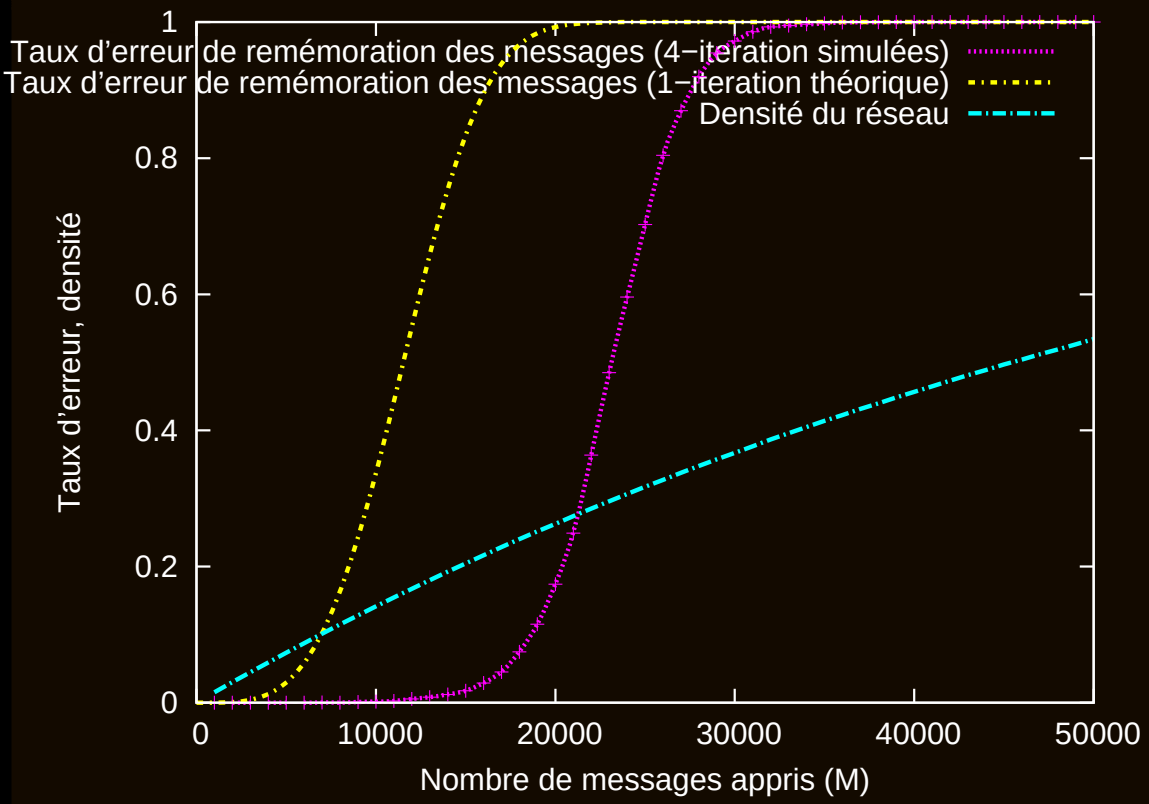
Corrélation *or not* corrélation





Performances

Mémoire associative

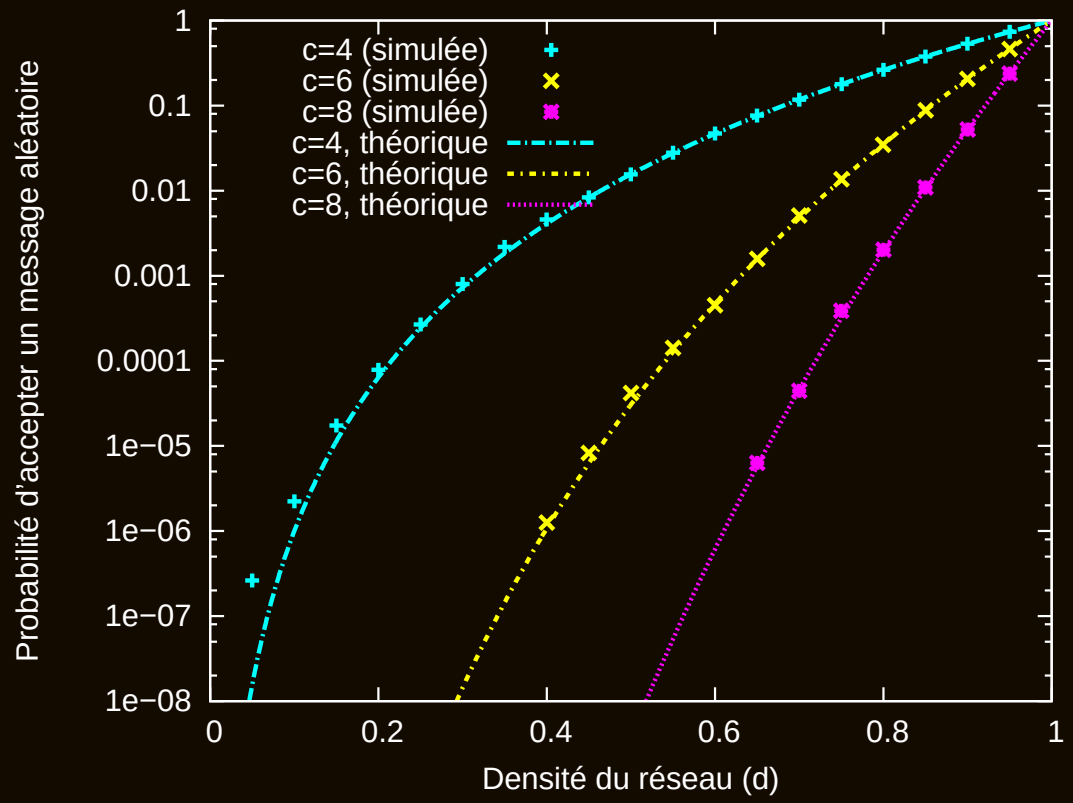


Probabilité d'erreur lors de la remémoration de messages appris avec la moitié (4) des 8 clusters non informés et pour $l = 256$.

- Gains par rapport au réseau de Hopfield : 130 en diversité, 12 en capacité, et 11 en efficacité (4.9% $\rightarrow$ 53.3%). Les performances dépendent principalement de l.

Performances (suite)

Classification

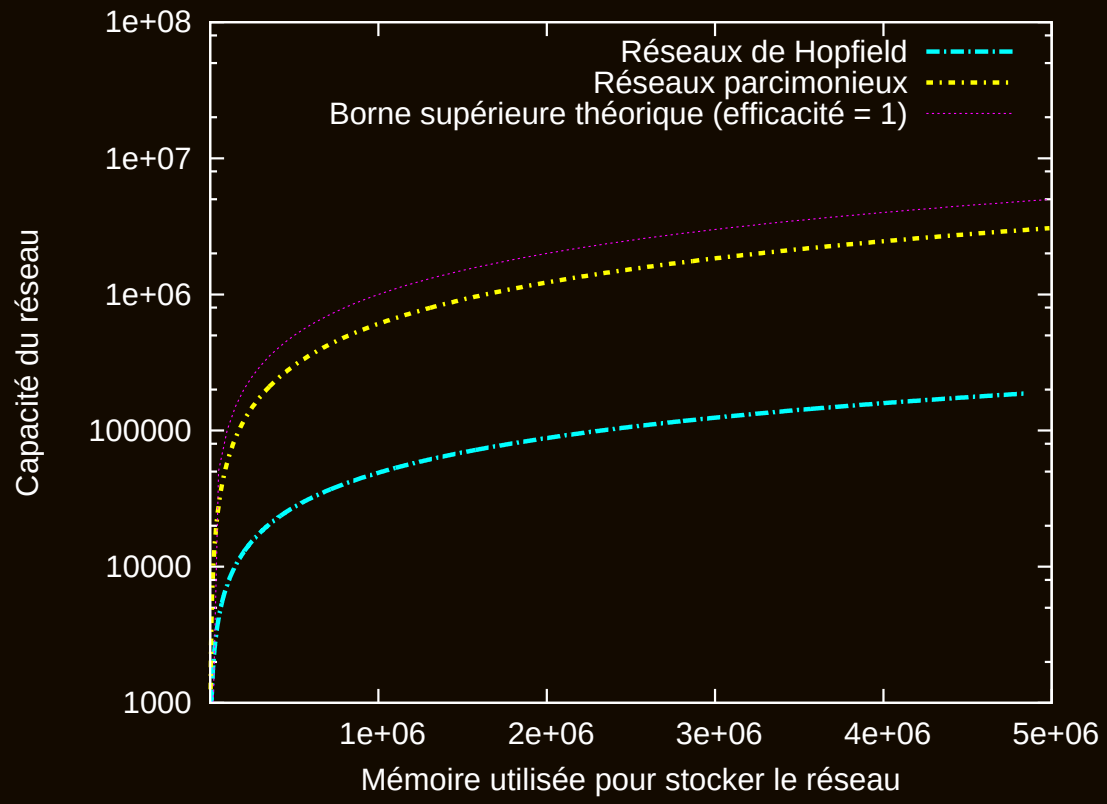


Probabilité d'accepter un message aléatoire (erreur de seconde espèce) pour des clusters de taille $l = 512$.

- Pas d'erreur de première espèce,
- Une erreur de seconde espèce très bonne, qui dépend de c.

Comparaison en capacité

Courbes de capacité



Capacité des réseaux de Hopfield et des réseaux de neurones parcimonieux en fonction de la quantité de mémoire utilisée pour stocker les réseaux. Pour les réseaux parcimonieux : $c = 16$ et la probabilité d'erreur en cas d'effacement d'un cluster est 0.01.

- Très proche de l'optimal,
- Gain énorme comparé à Hopfield.

Vers un quatrième niveau de parcimonie

Objectif

Faire croître les performances sans faire croître l.

1,2,3...et 4

- Messages de longueur $k \leq n$,
- Un unique représentant dans chaque cluster,
- Réseau creux...
- ...Messages parcimonieux.

Illustration

Vers un quatrième niveau de parcimonie

Objectif

Faire croître les performances sans faire croître l.

1,2,3...et 4

- Messages de longueur $k \leq n$,
- Un unique représentant dans chaque cluster,
- Réseau creux...
- ...Messages parcimonieux.

Illustration

Vers un quatrième niveau de parcimonie

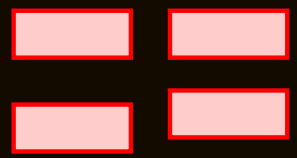
Objectif

Faire croître les performances sans faire croître l.

1,2,3...et 4

- Messages de longueur $k \leq n$,
- Un unique représentant dans chaque cluster,
- Réseau creux...
- ...Messages parcimonieux.

Illustration



Vers un quatrième niveau de parcimonie

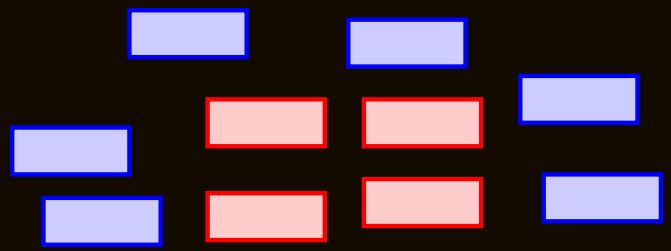
Objectif

Faire croître les performances sans faire croître l.

1,2,3...et 4

- Messages de longueur $k \leq n$,
- Un unique représentant dans chaque cluster,
- Réseau creux...
- ...Messages parcimonieux.

Illustration



Sous-réseaux et coïncidences

Parcimonie contrôlée

- Pour éviter l'épilepsie, la parcimonie doit être contrôlée,
- Par exemple, deux messages adressent soit les mêmes clusters soit au maximum 1 en commun,
- $\Rightarrow$ Les sous-réseaux adressés forment un code $C^{cc'}(c, 1)$,
- La densité est majorée par celle des sous-réseaux.

Coïncidences

- Chaque couple de cluster est caractéristique d'un sous-réseau,
- Ces couples sont colorés, par exemple par un temps caractéristique,
- Les neurones ne s'activent que s'il y a coïncidence temporelle,
- Le seuil σ permet d'éviter l'épilepsie.

Sous-réseaux et coïncidences

Parcimonie contrôlée

- Pour éviter l'épilepsie, la parcimonie doit être contrôlée,
- Par exemple, deux messages adressent soit les mêmes clusters soit au maximum 1 en commun,
- $\Rightarrow$ Les sous-réseaux adressés forment un code $C^{cc'}(c, 1)$,
- La densité est majorée par celle des sous-réseaux.

Coïncidences

- Chaque couple de cluster est caractéristique d'un sous-réseau,
- Ces couples sont colorés, par exemple par un temps caractéristique,
- Les neurones ne s'activent que s'il y a coïncidence temporelle,
- Le seuil σ permet d'éviter l'épilepsie.

Performances

Diversité

- Le réseau entièrement adressé apprend $\approx \alpha \left(\frac{n}{c}\right)^2$ mots,
- Si l'on a multiplié le nombre de clusters par c', on a :
  - c'^2 sous-réseaux de c clusters,
  - Chacun apprend $\approx \alpha \left(\frac{n}{cc'}\right)^2$ messages,
- Au final $\approx \alpha \left(\frac{n}{c}\right)^2$.

Sur le recouvrement plus grand que 1

- Si on autorise $\alpha \geq 1$ recouvrements, le nombre de sous-réseaux devient $c'^{\alpha+1}$,
- En contrepartie, la densité dépasse celle des sous-réseaux,
- Question ouverte sur le nombre optimal de recouvrements.

Performances

Diversité

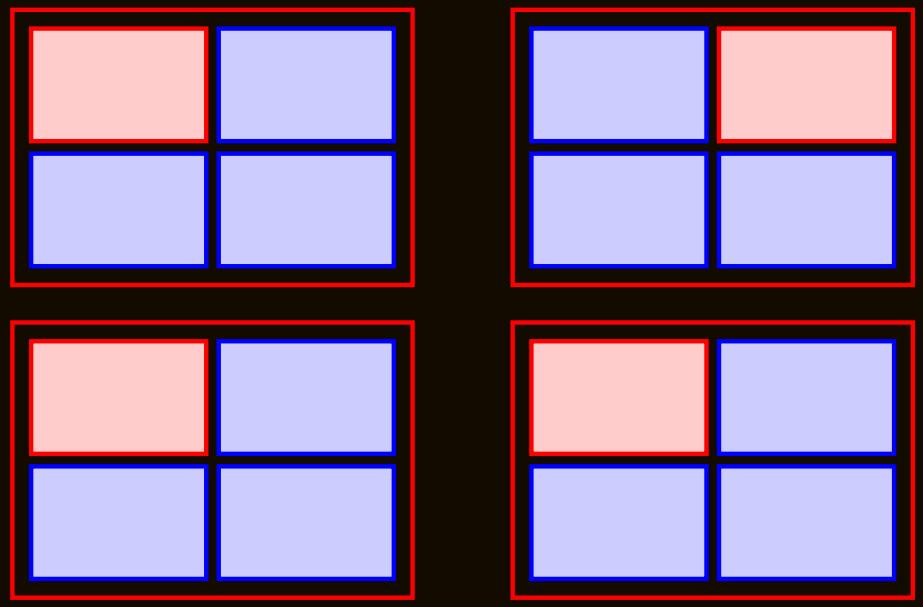
- Le réseau entièrement adressé apprend $\approx \alpha \left(\frac{n}{c}\right)^2$ mots,
- Si l'on a multiplié le nombre de clusters par c', on a :
  - c'^2 sous-réseaux de c clusters,
  - Chacun apprend $\approx \alpha \left(\frac{n}{cc'}\right)^2$ messages,
- Au final $\approx \alpha \left(\frac{n}{c}\right)^2$.

Sur le recouvrement plus grand que 1

- Si on autorise $\alpha \geq 1$ recouvrements, le nombre de sous-réseaux devient $c'^{\alpha+1}$,
- En contrepartie, la densité dépasse celle des sous-réseaux,
- Question ouverte sur le nombre optimal de recouvrements.

Parcimonie fractale

Winner take all entre clusters



Performances, remarques

- Mêmes performances dans les mêmes conditions,
- Gain de performances si effacement partiel,
- Les messages appris codent leur propre emplacement physique.

Note sur les espaces corrélés

Problème

L'apprentissage exploite une corrélation choisie, mais souffre de la corrélation subie.

Corrélation dans l'espace d'apprentissage

matin et *malin* appris $\rightarrow$ ambiguïté en cas d'effacement.

Corrélation dans le modèle proposé

- Si le réseau apprend *aaa*, *abb* et *bab*, il apprend aussi *aab*...
- Idée : ajouter des signatures cachées et aléatoires associées aux mots appris.

Exemple

- Les mots de la langue française de 6 lettres,
- Les performances passent de 30% de réussite à 80%.

Note sur les espaces corrélés

Problème

L'apprentissage exploite une corrélation choisie, mais souffre de la corrélation subie.

Corrélation dans l'espace d'apprentissage

matin et *malin* appris $\rightarrow$ ambiguïté en cas d'effacement.

Corrélation dans le modèle proposé

- Si le réseau apprend *aaa*, *abb* et *bab*, il apprend aussi *aab*...
- Idée : ajouter des signatures cachées et aléatoires associées aux mots appris.

Exemple

- Les mots de la langue française de 6 lettres,
- Les performances passent de 30% de réussite à 80%.

Note sur les espaces corrélés

Problème

L'apprentissage exploite une corrélation choisie, mais souffre de la corrélation subie.

Corrélation dans l'espace d'apprentissage

matin et *malin* appris $\rightarrow$ ambiguïté en cas d'effacement.

Corrélation dans le modèle proposé

- Si le réseau apprend *aaa*, *abb* et *bab*, il apprend aussi *aab*...
- Idée : ajouter des signatures cachées et aléatoires associées aux mots appris.

Exemple

- Les mots de la langue française de 6 lettres,
- Les performances passent de 30% de réussite à 80%.

Note sur les espaces corrélés

Problème

L'apprentissage exploite une corrélation choisie, mais souffre de la corrélation subie.

Corrélation dans l'espace d'apprentissage

matin et *malin* appris $\rightarrow$ ambiguïté en cas d'effacement.

Corrélation dans le modèle proposé

- Si le réseau apprend *aaa*, *abb* et *bab*, il apprend aussi *aab*...
- Idée : ajouter des signatures cachées et aléatoires associées aux mots appris.

Exemple

- Les mots de la langue française de 6 lettres,
- Les performances passent de 30% de réussite à 80%.

Plausibilité, applications

Plausibilité biologique

- Neurones binaires, connexions binaires, forte résilience ($\neq$ Hopfield),
- Faible densité globale, forte intéraction locale (petit monde),
- Opérations biologiquement plausibles : somme et *winner takes all*,
- Fonctionnement par cluster, spécialisation des neurones...

Applications

- Mémoires associatives,
- Classification (*go no-go*),
- Tri,
- Association d'informations, idées, concepts... Des messages indépendants partagent le même support physique.

Plausibilité, applications

Plausibilité biologique

- Neurones binaires, connexions binaires, forte résilience ($\neq$ Hopfield),
- Faible densité globale, forte intéraction locale (petit monde),
- Opérations biologiquement plausibles : somme et *winner takes all*,
- Fonctionnement par cluster, spécialisation des neurones...

Applications

- Mémoires associatives,
- Classification (*go no-go*),
- Tri,
- Association d'informations, idées, concepts... Des messages indépendants partagent le même support physique.

Conclusion

Bilan

- Codage parcimonieux :
  - Gains importants sur la diversité d'apprentissage,
  - Amélioration nette de l'efficacité de la mémorisation,
- Codage réparti : plus de messages appris que de neurones fanaux (approche turbo),
- Plausibilité biologique,
- Perspectives dans la conception de machines "intelligentes",
- Applications immédiates : mémoires associatives et classification.

Travaux en cours

- Influence du bruit, reconnaissance de messages flous,
- Effacements partiels de clusters,
- Réseaux de réseaux.

Conclusion

Bilan

- Codage parcimonieux :
  - Gains importants sur la diversité d'apprentissage,
  - Amélioration nette de l'efficacité de la mémorisation,
- Codage réparti : plus de messages appris que de neurones fanaux (approche turbo),
- Plausibilité biologique,
- Perspectives dans la conception de machines "intelligentes",
- Applications immédiates : mémoires associatives et classification.

Travaux en cours

- Influence du bruit, reconnaissance de messages flous,
- Effacements partiels de clusters,
- Réseaux de réseaux.

Fin de l'exposé

Merci pour votre attention, je suis à votre disposition pour toute question complémentaire.