

CHAPITRE 3

STRATÉGIES D'APPRENTISSAGE STANDARD ÉVALUÉES SUR LES BASES DE DONNÉES SYNTHÉTIQUES

Ce chapitre présente les résultats obtenus avec les bases de données synthétiques. Ces bases sont utilisées pour mesurer les performances des réseaux FAM au niveau des performances en généralisation, des temps de convergence et des taux de compression. De plus, les simulations ont permis de confirmer qu'il y a bien du sur-apprentissage dans les réseaux FAM.

Ce chapitre traite de plusieurs aspects considérés lors des simulations, soit: les effets dus à la taille de la base d'apprentissage, à la structure des bases de données, à la technique de normalisation, au degré de chevauchement ainsi qu'à la polarité du MatchTracking.

Tous ces aspects sont étudiés, cas par cas, pour bien mesurer leurs impacts sur les performances en généralisation, les temps de convergence ainsi que sur les taux de compression des réseaux FAM. De plus, lors des expériences, les paramètres internes du FAM sont les paramètres généraux soit: $\alpha = 0.01$, $\bar{\rho} = 0.0$, $\beta : 1.0$ et $\varepsilon = 0.001$ (MT+) ou -0.001 (MT-). Les résultats obtenus dans ce chapitre ont contribué à la publication d'un article [34].

3.1 Effets de la taille de la base d'apprentissage

Pour mesurer les effets engendrés par la taille de la base de données d'apprentissage, celle-ci est graduellement augmentée de 5 à 5000 patrons par classe en respectant une progression logarithmique (voir tableau II, section 2.1.1). L'étude de cet effet nous permet de vérifier la dégradation des performances en fonction de la taille de la base d'apprentissage.

De plus, l'utilisation de quatre stratégies d'apprentissage permet de montrer l'impact du nombre d'époques d'entraînement sur les performances des réseaux FAM et ainsi voir si ce facteur peut engendrer une dégradation des performances du réseau FAM.

Cette section est divisée en trois sous-sections. La première section présente les effets de la taille de la base d'apprentissage sur les bases de données synthétiques avec chevauchement et la deuxième section présente ces effets sur les bases de données synthétiques sans chevauchement. La troisième partie présente l'analyse des effets engendrés par la taille de la base d'apprentissage.

3.1.1 Bases de données avec chevauchement

Deux bases de données synthétiques avec chevauchement sont utilisées, soit DB_{μ} et DB_{σ} . La Figure 9 présente l'erreur en généralisation, le taux de compression ainsi que le temps de convergence obtenus avec la base $DB_{\mu}(1\%)$ pour les quatre stratégies d'apprentissage en utilisant la technique de normalisation MinMax. La Figure 10 et la Figure 11 présentent respectivement les mêmes types de résultats pour les bases $DB_{\mu}(9\%)$ et $DB_{\mu}(25\%)$. Les résultats obtenus par les autres degrés de chevauchement de la base DB_{μ} sont présentés dans l'annexe 3.

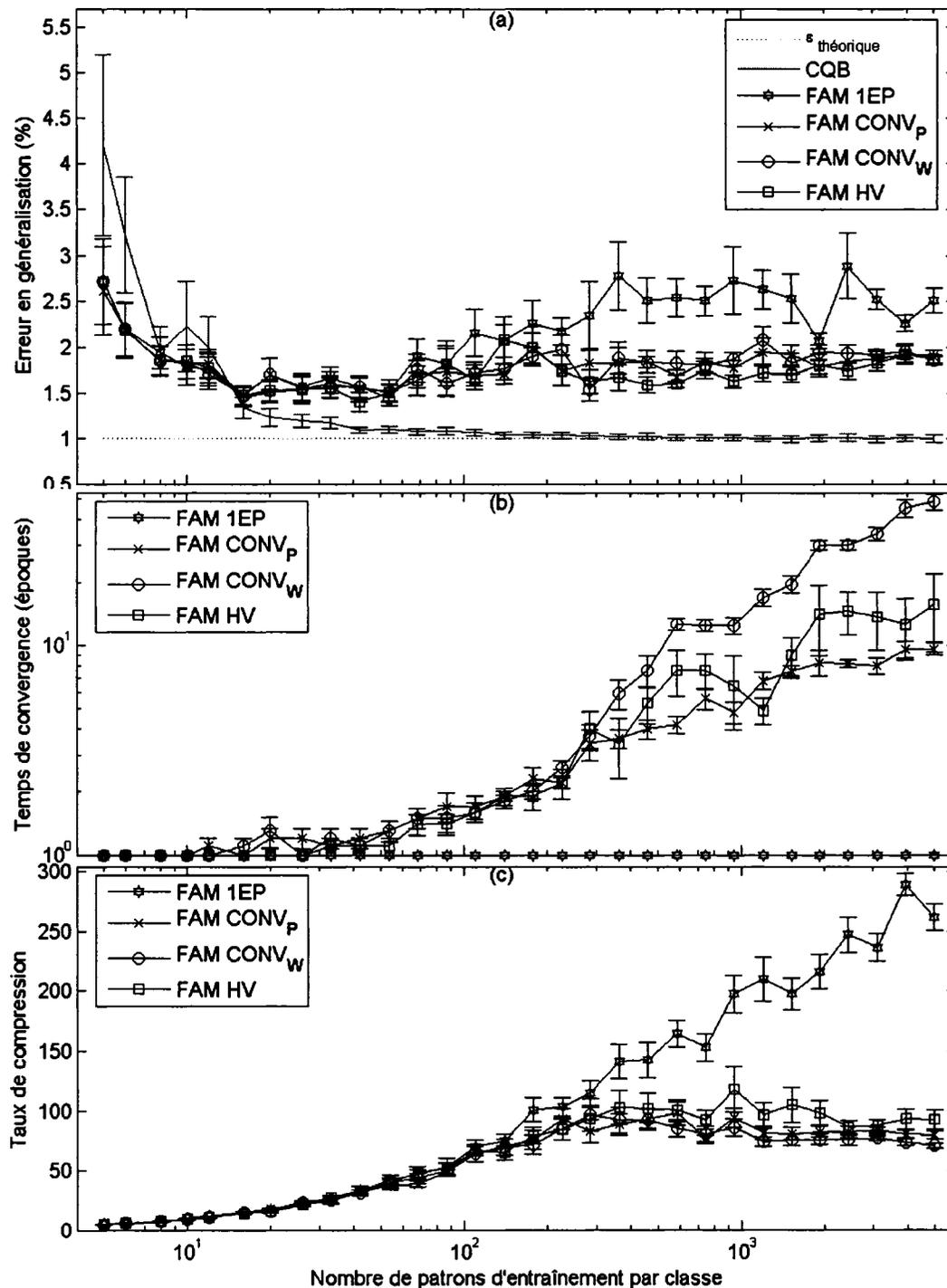


Figure 9 Performances moyennes du FAM en fonction de la taille de la base d'apprentissage avec la base $DB_{\mu}(1\%)$

(a) Erreur en généralisation, (b) temps de convergence et (c) taux de compression

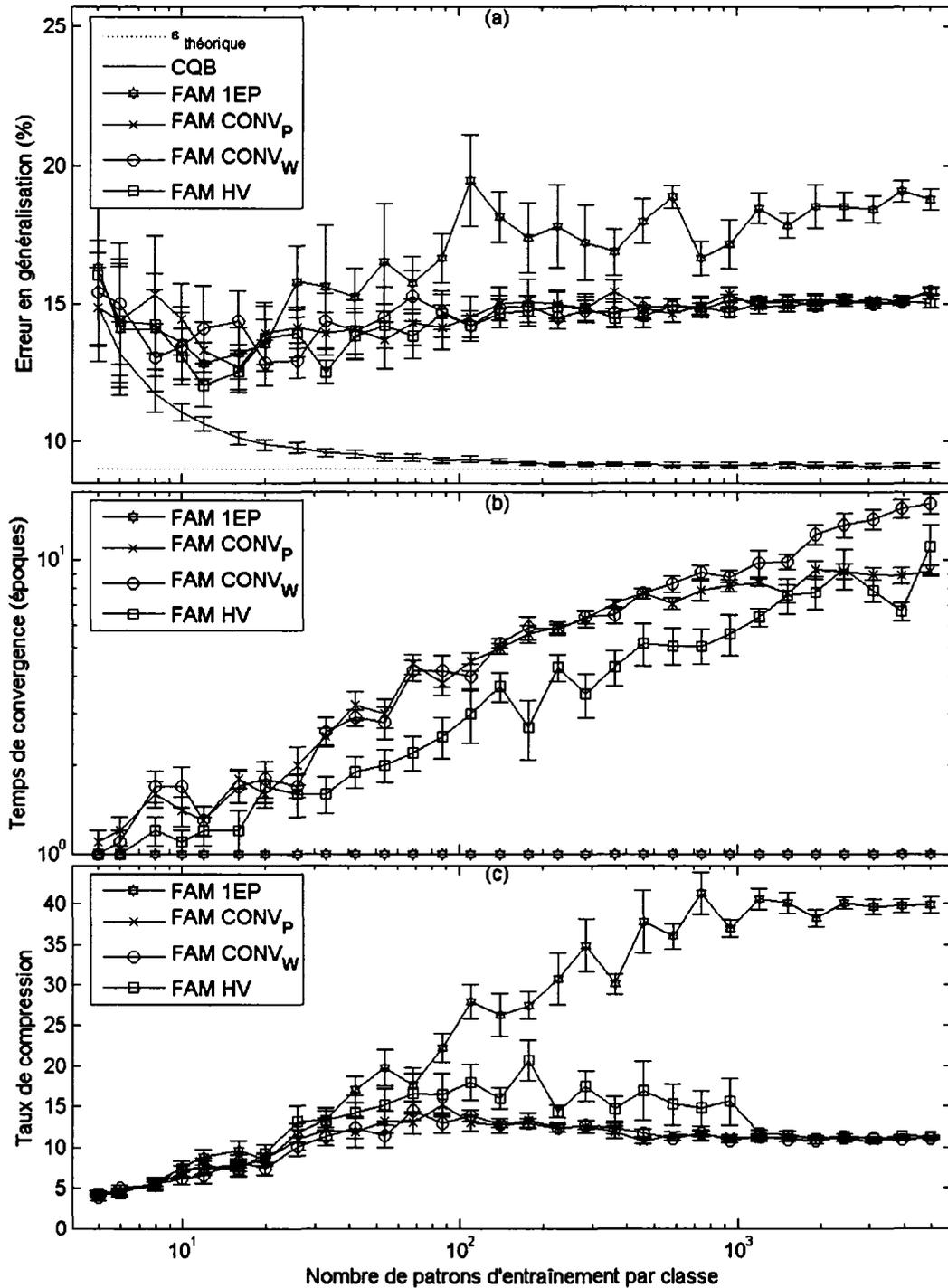


Figure 10 Performances moyennes du FAM en fonction de la taille de la base d'apprentissage avec la base $DB_{\mu}(9\%)$

(a) Erreur en généralisation, (b) temps de convergence et (c) taux de compression

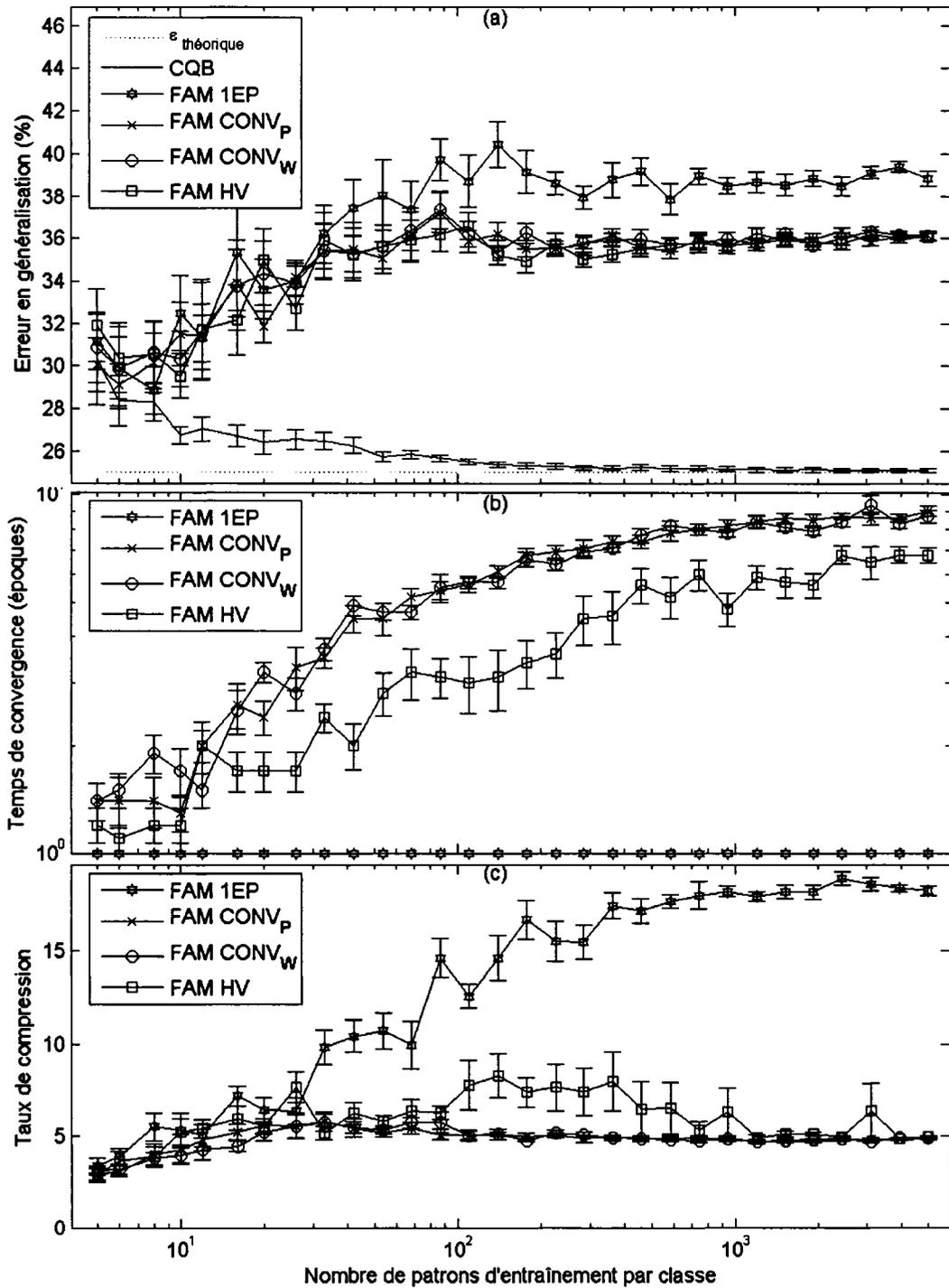


Figure 11 Performances moyennes du FAM en fonction de la taille de la base d'apprentissage avec la base $DB_{\mu}(25\%)$

(a) Erreur en généralisation, (b) temps de convergence et (c) taux de compression

3.1.2 Bases de données sans chevauchement

Deux bases de données sans chevauchement sont utilisées, soit DB_{CIS} et DB_{P2} . La Figure 12 présente les résultats généraux obtenus pour la base DB_{CIS} , soit l'erreur en généralisation, le temps de convergence ainsi que le taux de compression, pour les quatre stratégies d'apprentissage, lors de l'utilisation de la technique de normalisation MinMax. La Figure 13 présente les mêmes types de résultats généraux obtenus avec la base DB_{P2} .

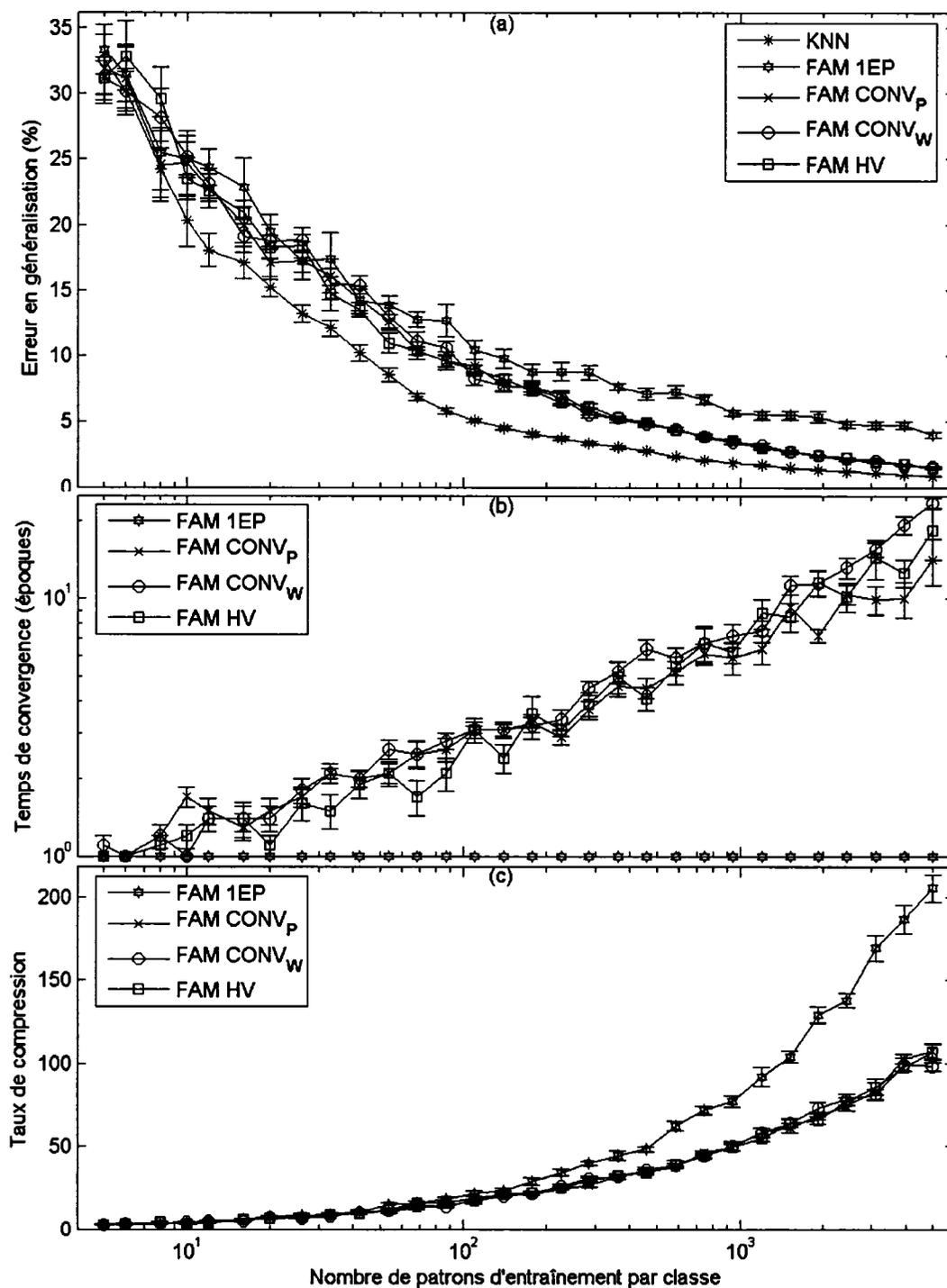


Figure 12 Performances moyennes du FAM en fonction de la taille de la base d'apprentissage avec la base DB_{CIS}

(a) Erreur en généralisation, (b) temps de convergence et (c) taux de compression

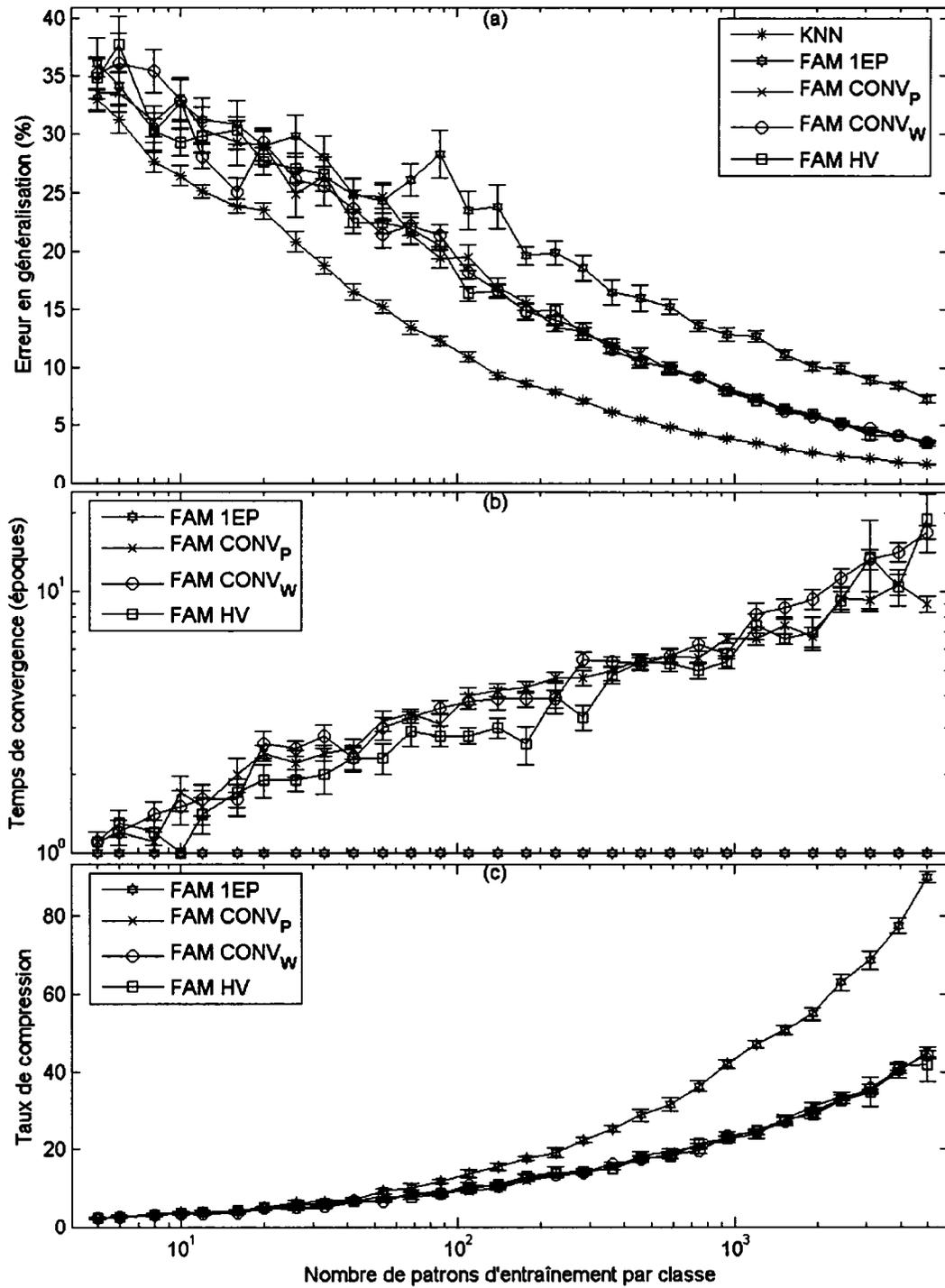


Figure 13 Performances moyennes du FAM en fonction du nombre de patrons d'entraînement avec la base DBP₂

(a) Erreur en généralisation, (b) temps de convergence et (c) taux de compression

3.1.3 Analyse

En observant les résultats obtenus avec les bases de données avec chevauchement, on constate que, peu importe la méthode d'apprentissage utilisée, l'erreur en généralisation des réseaux FAM augmente lorsque la taille de la base d'apprentissage dépasse un certain niveau. Ainsi, pour ces bases, il existe un effet de sur-apprentissage qui est fonction de la taille de la base d'entraînement.

La Figure 10(b) présente le taux de compression obtenu lors des tests avec la base de données $DB_{\mu}(9\%)$. On remarque que, lorsque les réseaux FAM commencent à subir une dégradation des performances en généralisation due à la taille de la base d'apprentissage, soit lorsque l'erreur en généralisation augmente, la compression des deux stratégies utilisant la convergence cesse d'augmenter.

Nous savons que la stratégie d'apprentissage HV est immunisée contre le phénomène de sur-apprentissage en fonction du nombre d'époques d'entraînement. Les stratégies $CONV_W$ et $CONV_P$ obtiennent des erreurs en généralisation légèrement supérieures ainsi que des temps de convergence supérieurs à la stratégie HV. En tenant compte de ces deux aspects, nous pouvons conclure que les réseaux fuzzy ARTMAP peuvent engendrer du sur-apprentissage en fonction du nombre d'époques d'entraînement. Cependant, plus la taille de la base d'apprentissage est grande, moins ce phénomène est apparent.

En regardant l'ensemble des résultats obtenus, en faisant varier la taille de la base d'apprentissage, on peut observer le phénomène suivant: dès qu'il y a chevauchement entre deux classes, il peut y avoir un effet de sur-apprentissage dû à la taille de la base d'apprentissage et au nombre d'époques d'apprentissage.

Analysons maintenant la manifestation de ce phénomène à l'intérieur des réseaux fuzzy ARTMAP. Lors de la phase d'apprentissage, la règle générale veut que le maximum de données soient utilisées. Cette règle s'appuie sur le concept que, plus il y a d'éléments dans la base d'apprentissage, meilleure est la représentation de la dispersion réelle des données. Cependant, les résultats obtenus démontrent que, dans certains cas, la performance des réseaux fuzzy ARTMAP chute lorsqu'il y a trop de données dans la base d'apprentissage. Ceci est dû à la prolifération des catégories, c'est-à-dire que le réseau final a créé trop de catégories lors de l'entraînement. Ce surplus de catégories diminue les performances qu'aurait pu atteindre le réseau et provient du fait que trop de patrons ont été présentés lors de la phase d'apprentissage.

La Figure 14(a) présente les bornes de décision (a.1) et les catégories (a.2) obtenues lors de l'entraînement avec la taille maximale de la base d'apprentissage (5000 patrons par classe) pour un des tests avec la base $DB_{\mu}(9\%)$ en utilisant la stratégie d'apprentissage HV.

Ce réseau FAM est composé de 916 catégories et obtient une erreur en généralisation de 14.85%, soit 5.85% de plus que l'erreur théorique fixe à 9%. Cette différence est due au grand nombre de petites zones en erreur, soit celles situées de l'autre côté de la frontière de décision optimale. Ces petites zones en erreur proviennent du fait que des catégories appartenant à la classe 1 sont totalement comprises dans une zone qui devrait statistiquement appartenir à la classe 2, et vice-versa. Il faut mentionner que les régions $[(0,0.3);(0.7,1)]$ et $[(0.7,1);(0,0.3)]$ n'influencent pratiquement pas la performance en généralisation, car un très faible nombre de données peuvent être situées dans ces deux régions.

La figure 14(b) présente les bornes de décision (b.1) et les catégories (b.2) obtenues avec 26 patrons d'apprentissage pour la même base de données et la même stratégie d'entraînement. Étant donné le faible nombre de patrons d'apprentissage utilisés, il n'y a

que 4 catégories créées lors de l'entraînement. Ce réseau obtient une erreur en généralisation plus faible qu'avec la taille maximum de la base d'apprentissage, soit de 9.87%, seulement 0.87% de plus que l'erreur théorique. Ceci crée une borne de décision beaucoup plus proche de la borne optimale et empêche la création des nombreuses petites zones en erreur tel que représenté par la Figure 14(a.1).

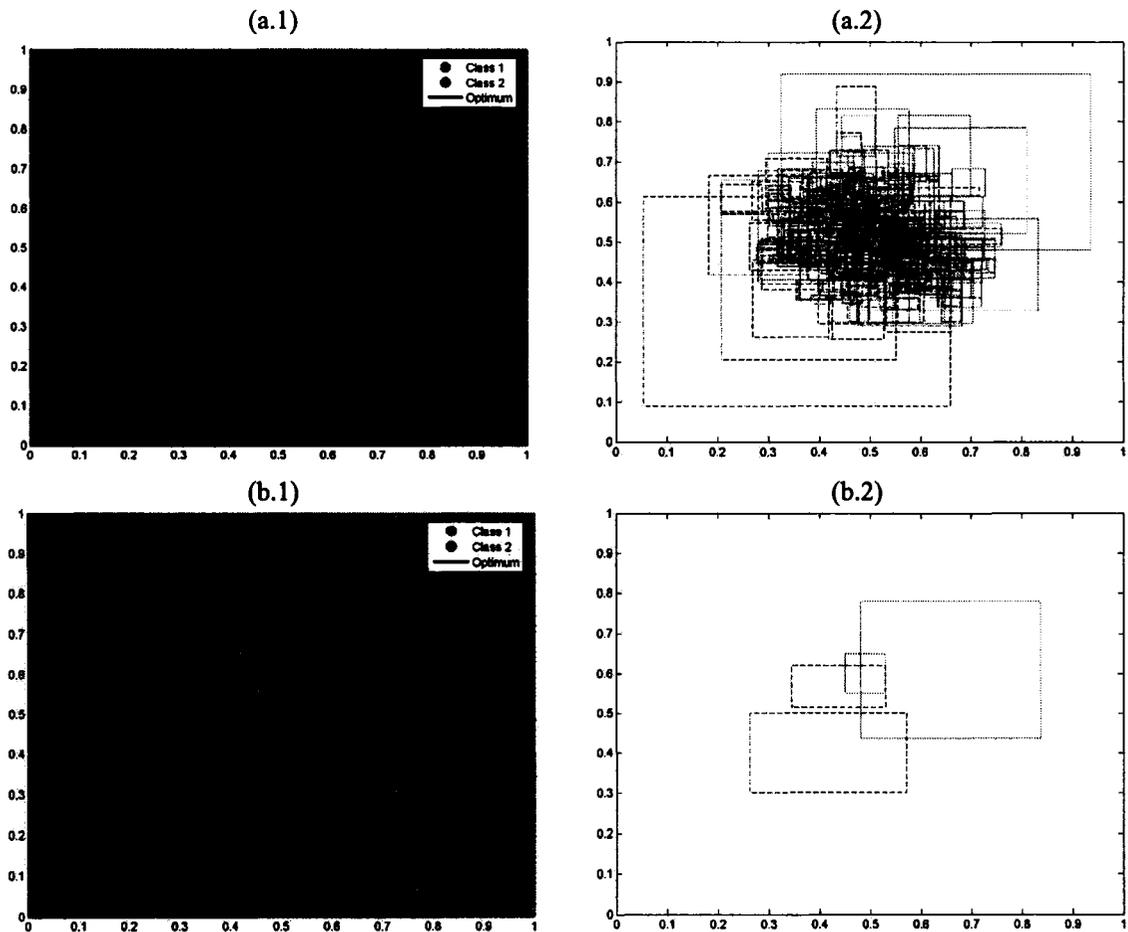


Figure 14 Catégories et bornes de décision obtenues pour $DB_{\mu}(9\%)$ avec HV
Soit (a) 5000 patrons et (b) 26 patrons d'entraînement par classe, et (.1) bornes de décision et (.2) catégories créées.

Ainsi, en optimisant la taille de la base d'apprentissage, sur un des tests avec la base $DB_{\mu}(9\%)$, nous avons réussi à passer d'une erreur en généralisation de 14.85% à 9.87%,

soit une réduction de 4.98%. Il y a donc un net avantage à optimiser la taille de la base d'entraînement pour les bases de données possédant un degré de chevauchement.

En observant les résultats obtenus avec les bases de données sans chevauchement (DB_{CIS} et DB_{P2}), il est évident qu'il n'y a aucune dégradation des performances du FAM en fonction de la taille de la base d'apprentissage. Ceci provient du fait qu'il n'y a pas de chevauchement entre les deux classes. Les nombreuses petites zones que nous avons observées avec les bases de données possédant un degré de chevauchement (voir figure 14) sont inexistantes car toutes les catégories créées sont totalement ou en partie dans leur zone optimale.

La Figure 15 présente les bornes de décision ainsi que les catégories créées avec la base de données DB_{CIS} lors de l'augmentation progressive du nombre de patrons d'entraînement, soit 10, 178 et 5000 patrons par classe. La stratégie d'apprentissage HV ainsi que la technique de normalisation MinMax sont utilisées. On constate que, plus la taille de la base d'apprentissage augmente, plus le nombre de catégories dans la région de transition entre les deux classes augmente, créant ainsi une borne de décision de plus en plus proche de la borne optimale. Cette augmentation de la base de données fait passer l'erreur en généralisation de 30.00% (10 patrons par classe), à 7.64% (178 patrons par classe) et finalement à 1.66% (5000 patrons par classe). On observe également le même phénomène avec la base de donnée DB_{P2} . Ainsi, la prolifération des catégories est une bonne chose dans le cas des distributions sans chevauchement car elle raffine la borne de décision pour ce type de problèmes.

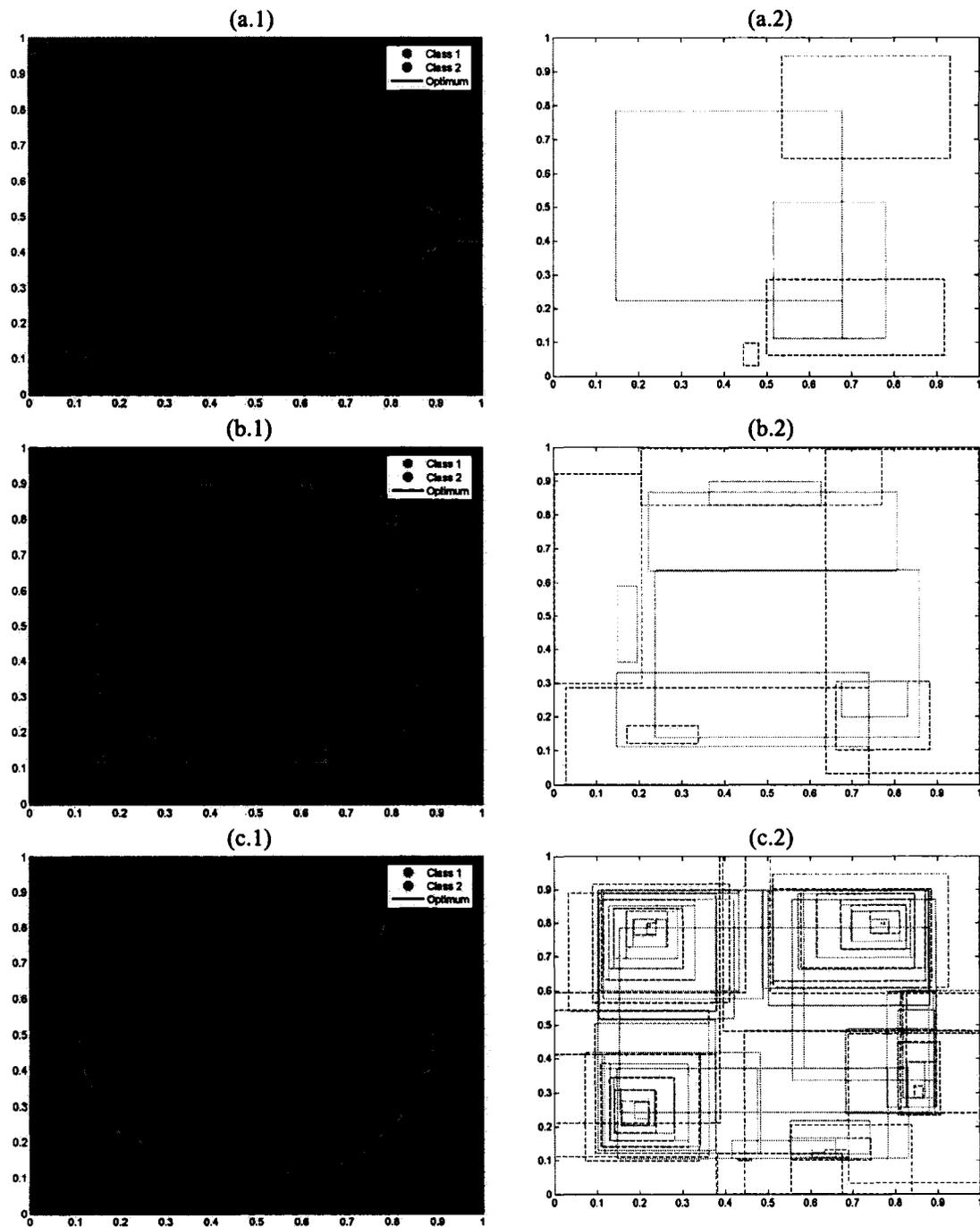


Figure 15 Bornes et catégories obtenues lors de l'accroissement de la taille de la base d'apprentissage avec la base DB_{CIS} .

Soit (a) 10 patrons, (b) 178 patrons et (c) 5000 patrons d'entraînement par classe, et (.1) bornes de décision et (.2) catégories créées.

De plus, aucune dégradation des performances du réseau FAM en fonction du nombre d'époques d'entraînement n'est visible dans les résultats des bases DB_{CIS} et DB_{P2} . En effet, bien que les temps de convergence entre les stratégies d'apprentissage ne soient pas identiques, il n'y a pas de différences significatives entre les performances en généralisation et les taux de compression obtenus par les diverses stratégies d'apprentissage, mis à part l'entraînement sur une époque (1EP).

Bref, peu importe qu'il y ait ou non du chevauchement entre les classes, la stratégie d'apprentissage 1EP génère, en moyenne, le taux de compression le plus élevé ainsi que la plus grande erreur en généralisation. La stratégie d'apprentissage HV donne, en moyenne, des résultats meilleurs ou équivalents aux deux stratégies basées sur la convergence des poids synaptiques et des patrons.

Pour mesurer les effets dus au degré de chevauchement entre les classes, celui-ci est progressivement augmenté de 1% à 25% (voir tableau III et Tableau IV, section 2.1.1) pour les deux bases DB_{μ} et DB_{σ} . Puisque nous venons de démontrer que le sur-apprentissage en fonction de la taille de la base d'entraînement peut exister dans les réseaux fuzzy ARTMAP, pour les bases de données avec chevauchement, nous pouvons présenter son évolution en fonction du degré de chevauchement. Pour ce faire, nous allons analyser l'impact du degré de chevauchement avec les quatre stratégies d'apprentissage.

Définissons l'erreur de sur-apprentissage comme étant la différence entre l'erreur en généralisation obtenue avec la grandeur maximale de la base d'apprentissage (5000 patrons par classe) et l'erreur en généralisation minimale obtenue sur l'ensemble de la taille de la base d'apprentissage. En appliquant l'équation (3.1), nous obtenons la moyenne de l'erreur de sur-apprentissage E_{sapp} évaluée sur l'ensemble des simulations que nous avons effectuées.

$$E_{sapp} = \frac{\sum_{i=1}^N E_{gen}(5k/\omega)_i - E_{gen}(\min)_i}{N} \quad (3.1)$$

Où : N est le nombre de répétitions lors des expérimentations (10)

$E_{gen}(5k/\omega)_i$ est l'erreur en généralisation obtenue avec 5000 patrons d'entraînement par classe, soit la taille maximale de la base d'apprentissage, de la $i^{ème}$ répétition.

$E_{gen}(\min)_i$ est l'erreur en généralisation minimale obtenue sur l'ensemble des tailles de la base d'apprentissage de la $i^{ème}$ répétition.

La Figure 16 présente la moyenne des erreurs de sur-apprentissage obtenue lors des tests avec les diverses bases de données DB_{μ} .

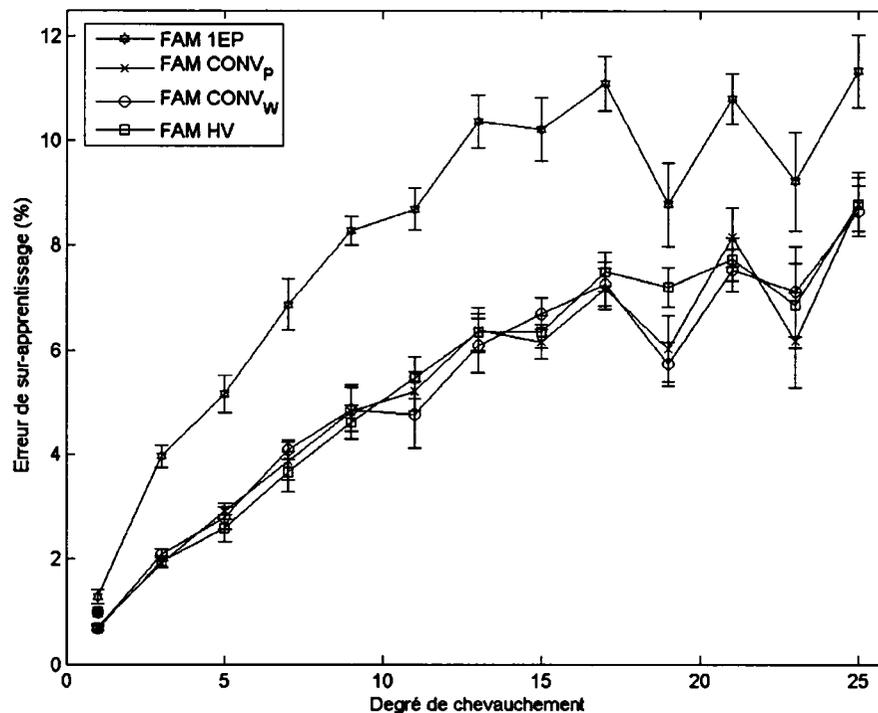


Figure 16 Erreurs de sur-apprentissage dues à la taille de la base d'apprentissage en fonction du degré de chevauchement

On remarque que plus le degré de chevauchement est grand, plus la taille de la base d'apprentissage joue un rôle important dans la dégradation des performances. De plus, nous avons constaté que plus le degré de chevauchement est grand, plus la taille optimale de la base d'entraînement est petite et vice-versa. Ainsi, avec une base de données sans chevauchement, il faut utiliser le maximum de données d'entraînement pour obtenir les meilleures performances en généralisation.

Les stratégies d'apprentissage $CONV_w$, $CONV_p$ et la HV génèrent des erreurs de sur-apprentissage semblables. Ainsi, elles obtiennent le même type d'écart entre les erreurs en généralisation minimales obtenues lors des répétitions, et les erreurs en généralisations obtenues avec la taille maximale de la base d'entraînement.

En regardant les résultats de la Figure 10, on pourrait conclure que l'erreur de sur-apprentissage pour la base $DB_\mu(9\%)$, lors de l'utilisation de la stratégie HV, ne devrait pas dépasser 3%. Pourtant, la Figure 16 montre qu'il y a environ 4.5% d'erreur de sur-apprentissage avec ce degré de chevauchement. Cet écart provient du fait que cette erreur est calculée individuellement sur chacune des 10 réplifications et que la taille optimale de chacune, bien que située dans la même région, n'est pas la même. Ainsi, même si l'on peut faire une approximation de l'erreur de sur-apprentissage avec la Figure 10, elle ne peut pas servir à établir convenablement cette erreur car elle représente la moyenne obtenue sur chaque taille et non les courbes individuelles des 10 réplifications.

Définissons également l'erreur nette comme étant la différence entre l'erreur en généralisation obtenue pour chaque répétition et l'erreur théorique de la base de données. Cette valeur indique l'éloignement du réseau FAM face à l'erreur théorique.

La Figure 17 présente l'erreur nette obtenue par chaque stratégie d'apprentissage lors de l'optimisation de la taille de la base d'entraînement pour chaque réplification. On

remarque que plus le degré de chevauchement est élevé, plus l'erreur nette est grande. De plus, les quatre stratégies d'apprentissage obtiennent des erreurs nettes semblables pour un degré de chevauchement donné.

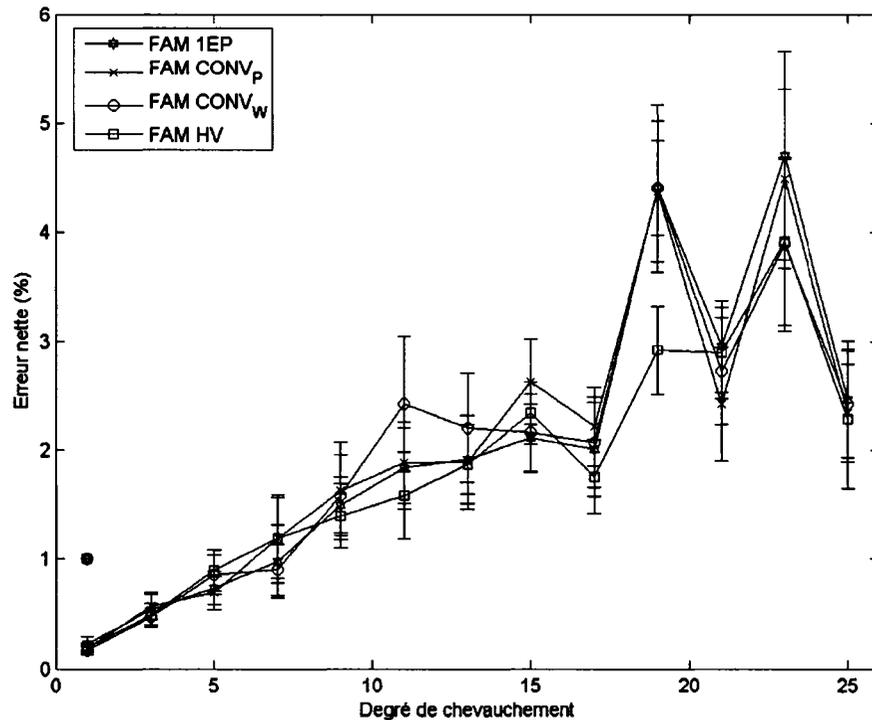


Figure 17 Erreur nette obtenue en sélectionnant la meilleur taille de la base d'apprentissage en fonction du degré de chevauchement

Pour démontrer à quel point il est important d'optimiser la taille de la base d'apprentissage pour les bases de données possédant un degré de chevauchement, la Figure 18 présente une comparaison entre l'erreur nette obtenue lors de l'optimisation de la taille de la base d'apprentissage et celle obtenue avec la taille maximale de la base d'apprentissage, soit 5000 patrons par classe; le tout pour la stratégie HV.

Cette figure démontre bien à quel point la taille de la base d'entraînement joue un rôle important dans la dégradation des performances des réseaux fuzzy ARTMAP lors de l'utilisation de bases avec chevauchement.

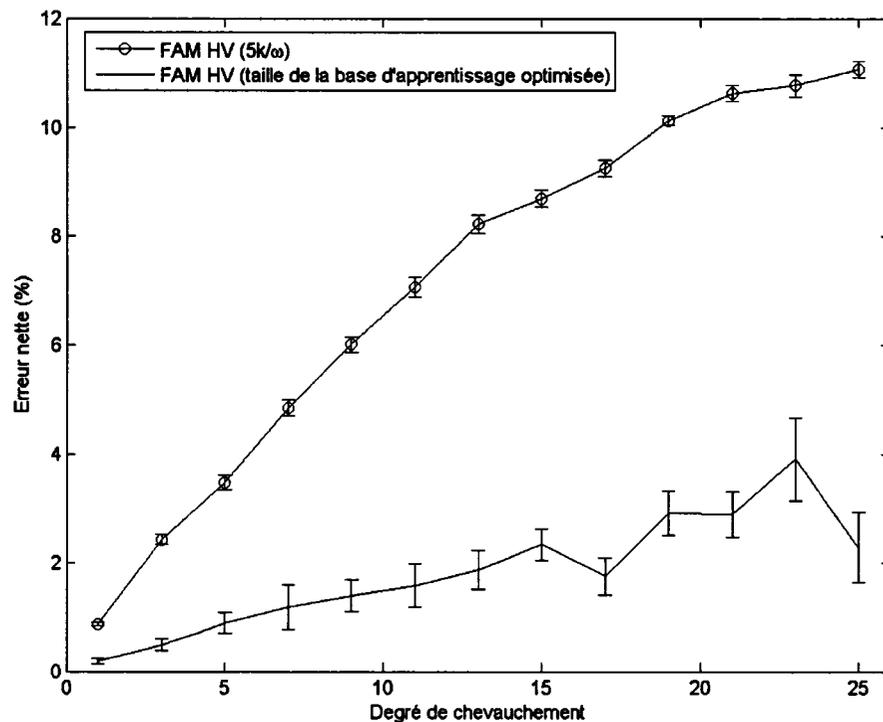


Figure 18 Erreur nette avec et sans optimisation du nombre de patrons d'apprentissage

3.2 Effets des structures des bases de données

Il existe plusieurs méthodes pour créer une base de données possédant un degré de chevauchement donné. Deux méthodes différentes ont été utilisées lors de la création des deux bases de données contenant du chevauchement. La première base, DB_{μ} , consiste en deux distributions gaussiennes dont la moyenne de la deuxième distribution est graduellement rapprochée de la moyenne de la première, augmentant ainsi le degré de chevauchement entre les distributions. La deuxième base, DB_{σ} , consiste en deux

distributions gaussiennes dont la variance des deux distributions est graduellement augmentée pour amplifier le degré de chevauchement entre les deux classes. Il résulte que, pour un taux d'erreur total identique, la frontière de décision entre les classes est plus grande avec DB_{σ} . Les paramètres utilisés lors de la création de ces deux bases de données sont présentés par le tableau III et le Tableau IV (section 2.1.1). Deux méthodes différentes ont également été utilisées lors de la création des deux bases de données sans chevauchement (DB_{CIS} et DB_{P2}).

3.2.1 Bases de données avec chevauchement

Cette section traite des différences obtenues entre les résultats des bases ayant un même degré de chevauchement mais dont la méthode de création diffère. Les figures 18 à 20 présentent ces différences pour une base de données de 9% de chevauchement. Les courbes présentes dans ces figures montrent les différences entre les résultats de DB_{μ} et de DB_{σ} . Ainsi, lorsque la courbe est positive, la valeur obtenue avec la base DB_{μ} est plus grande que celle obtenue avec la base DB_{σ} , et vice-versa. Ces figures présentent ces différences pour l'erreur en généralisation, le taux de compression ainsi que pour le temps de convergence. La technique de normalisation MinMax est utilisée. L'annexe 4 présente ces résultats pour les autres degrés de chevauchement utilisés.

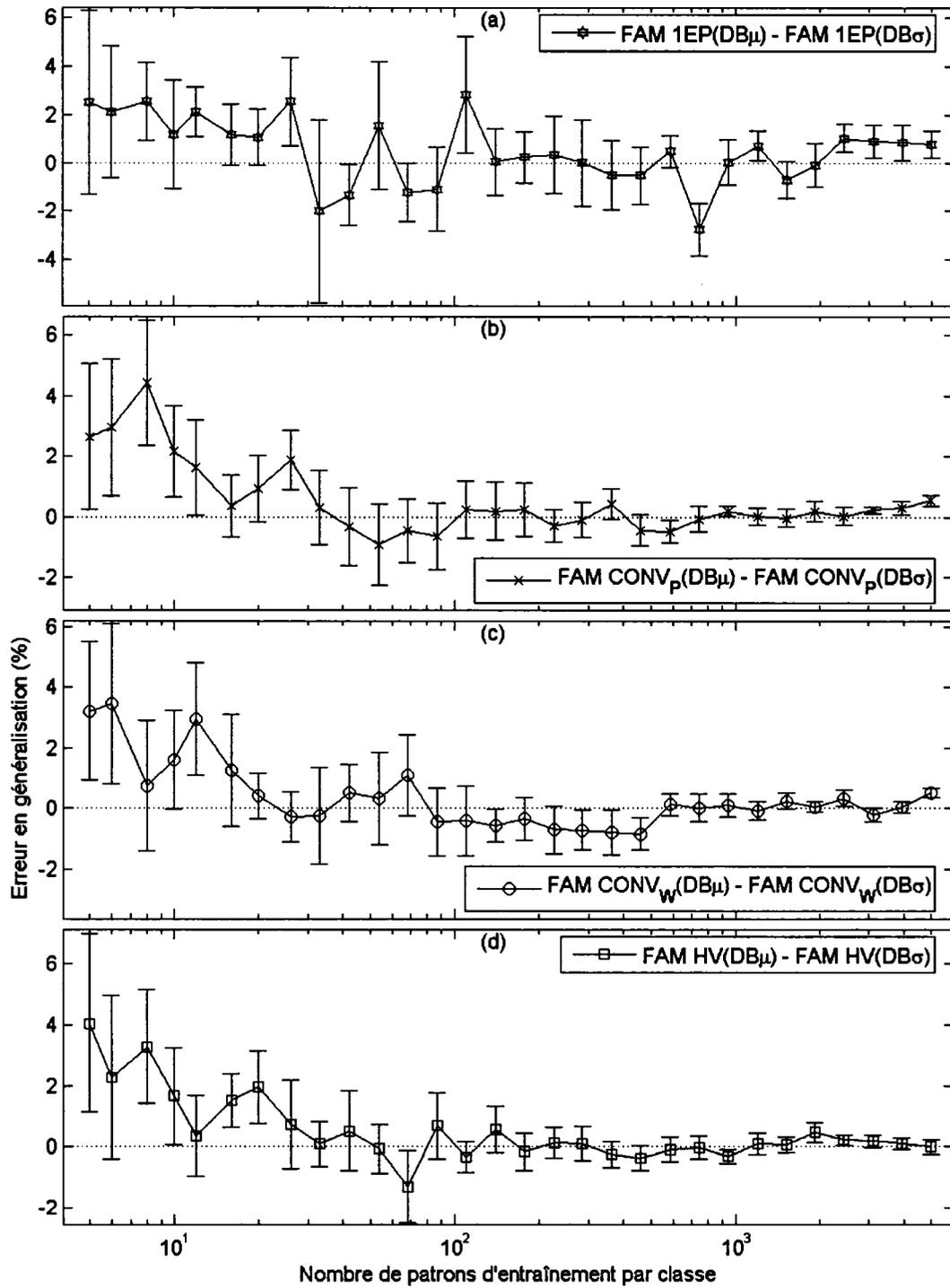


Figure 19 Différence entre $DB_\mu(9\%)$ et $DB_\sigma(9\%)$ sur l'erreur en généralisation
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

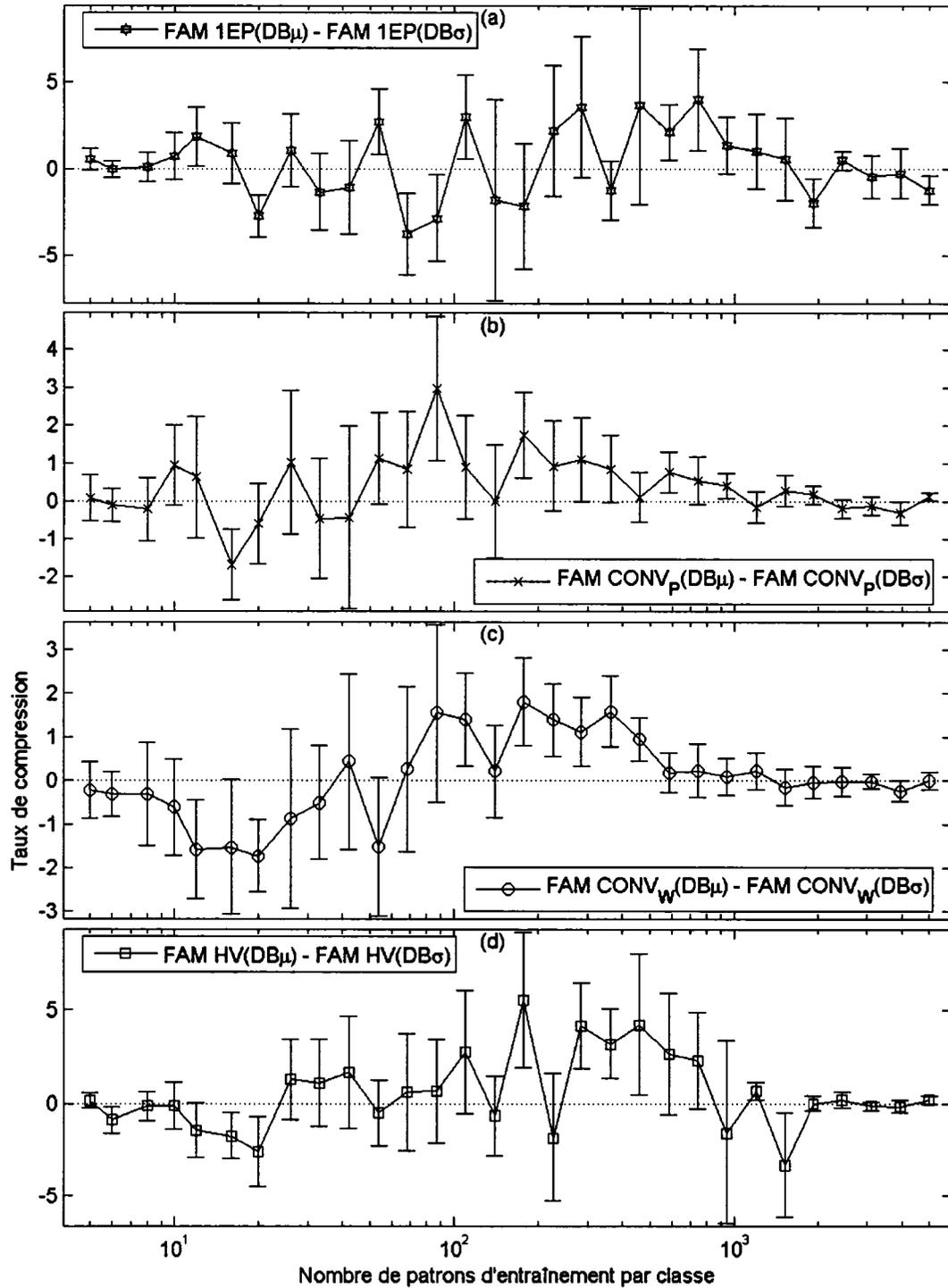


Figure 20 Différence entre $DB_{\mu}(9\%)$ et $DB_{\sigma}(9\%)$ sur le taux de compression
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

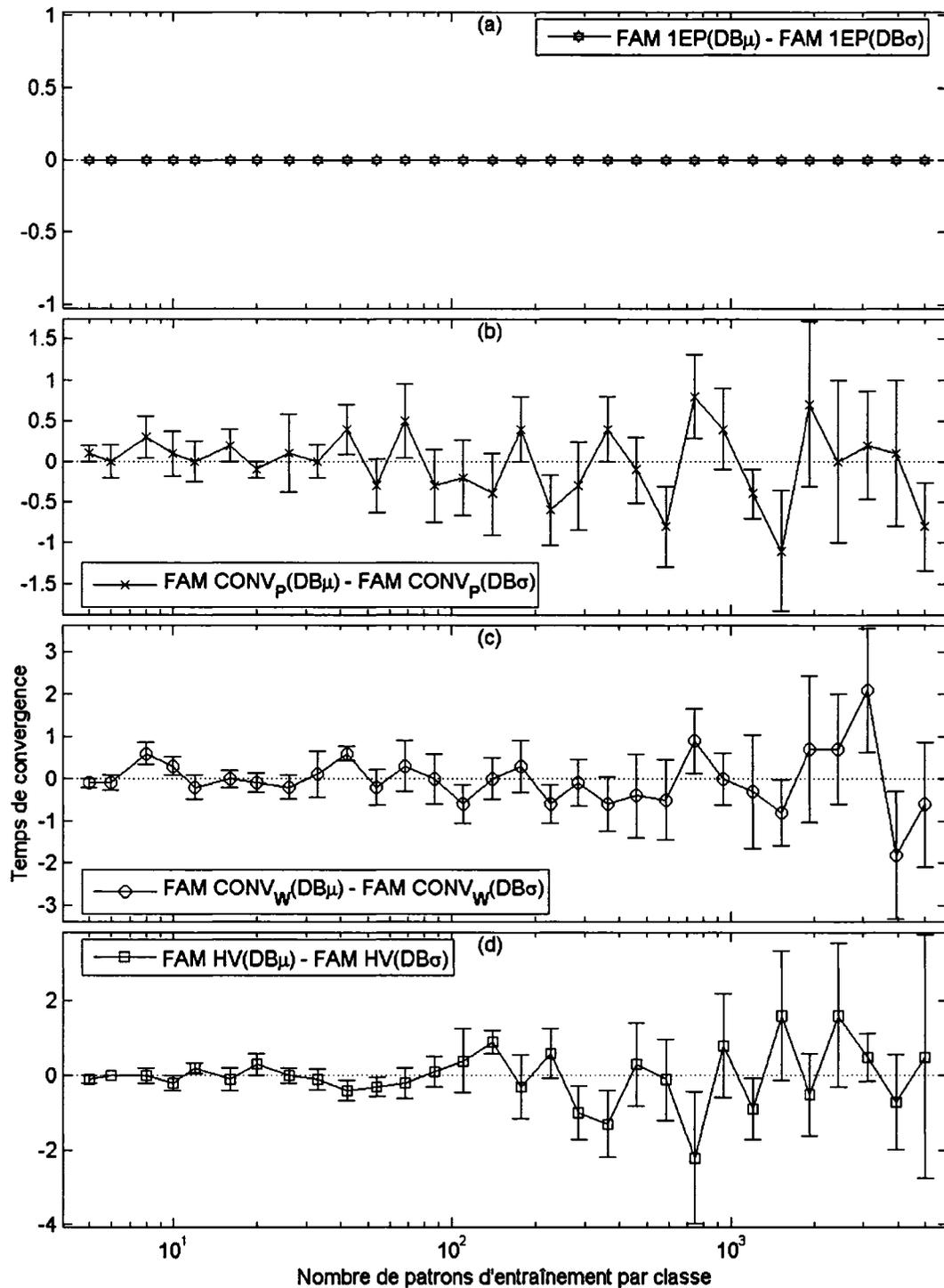


Figure 21 Différence entre $\text{DB}_\mu(9\%)$ et $\text{DB}_\sigma(9\%)$ sur le temps de convergence
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

3.2.2 Bases de données sans chevauchement

Bien que les deux bases de données sans chevauchement utilisées aient la même erreur théorique, elles ne représentent pas pour autant le même niveau de difficulté. En effet, la base DB_{P2} possède plusieurs frontières de décision d'un degré de complexité élevé comparativement à l'unique frontière de décision de la base DB_{CIS} .

Les figures 21 à 23 présentent une comparaison entre les deux problèmes DB_{P2} et DB_{CIS} pour les quatre stratégies d'apprentissage en fonction de la taille de la base d'apprentissage. La normalisation MinMax est utilisée.

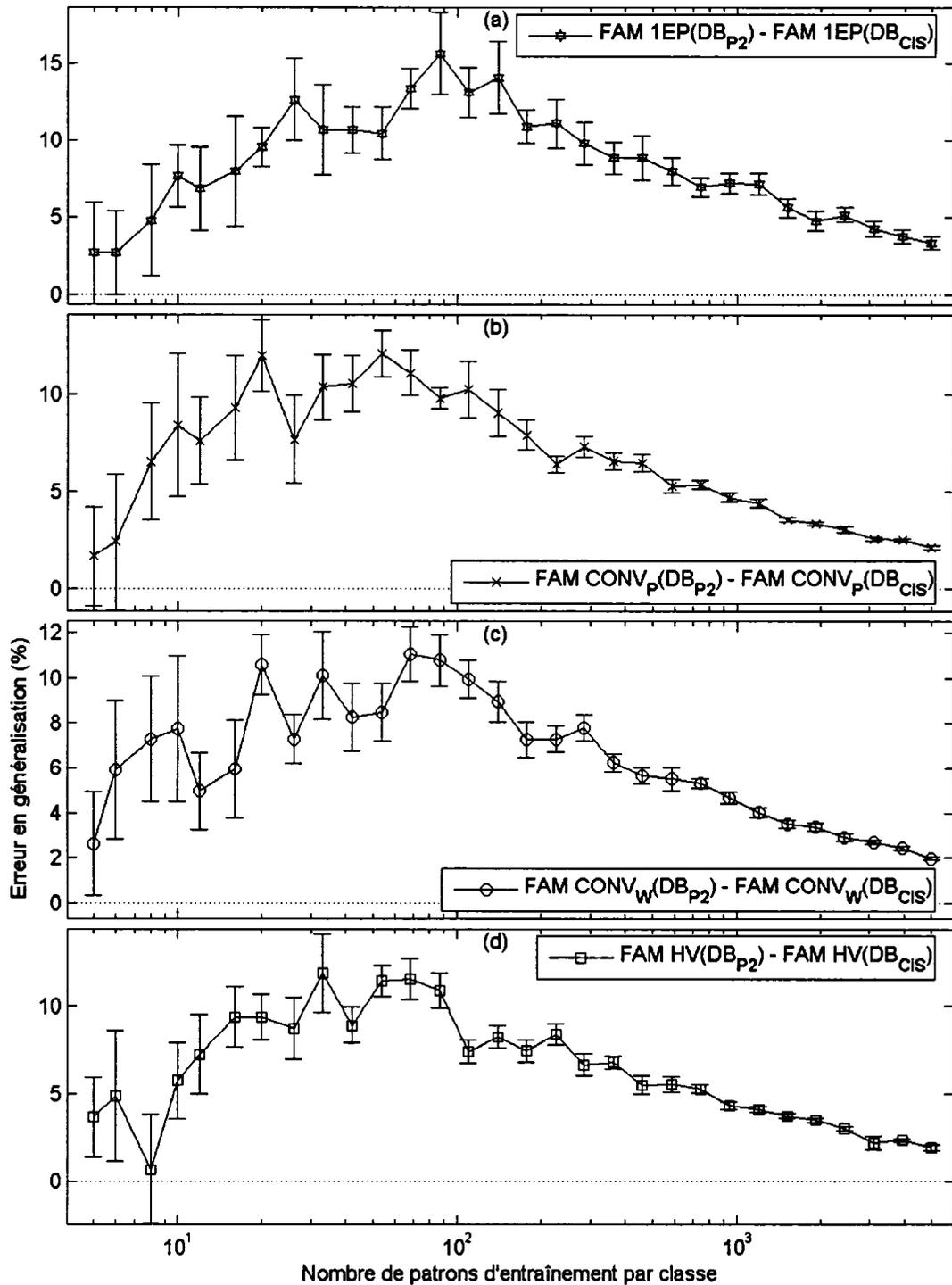


Figure 22 Différence entre DB_{CIS} et DB_{P2} sur l'erreur en généralisation
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

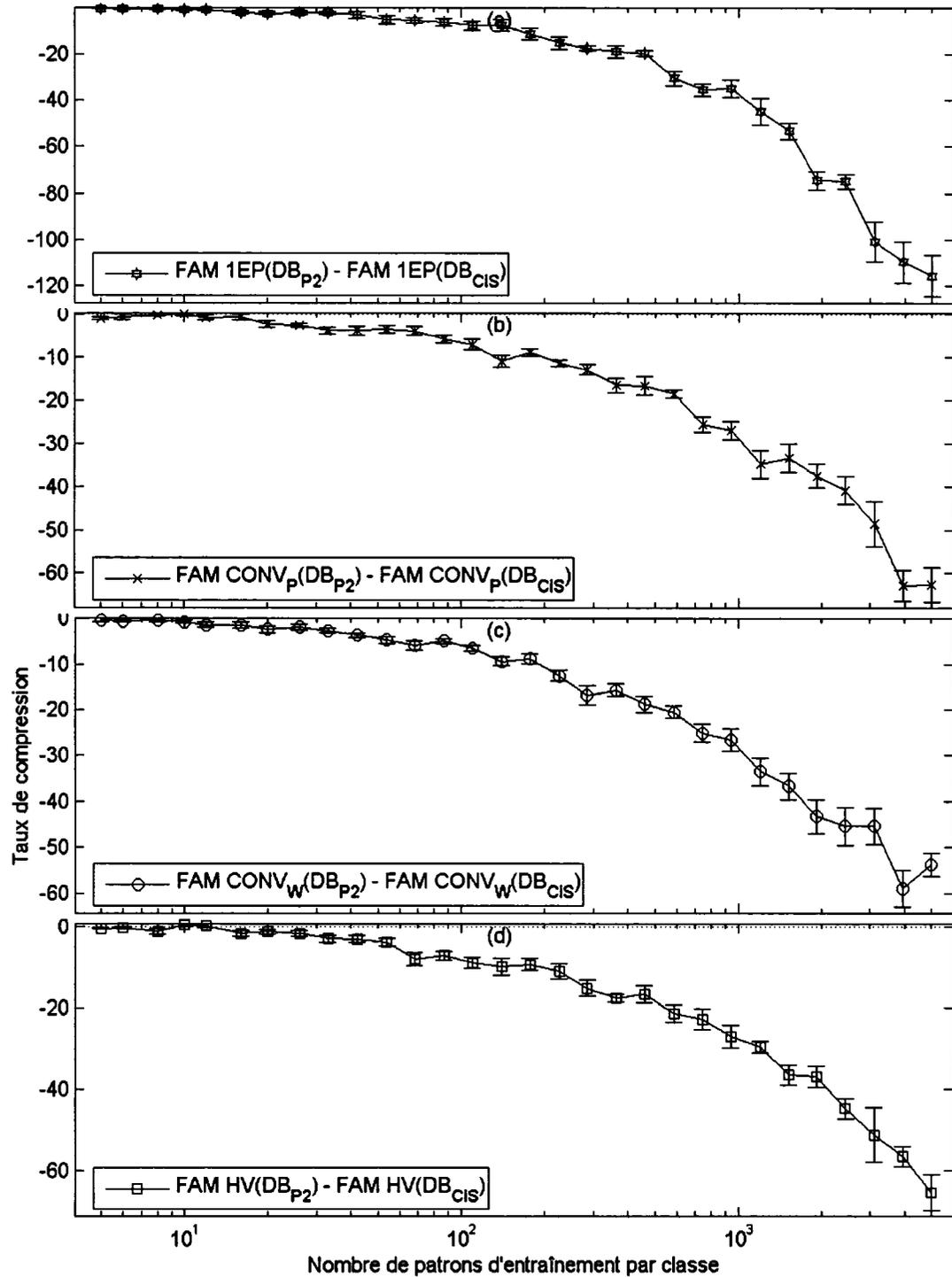


Figure 23 Différence entre DB_{CIS} et DB_{P2} sur le taux de compression

- (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et (d) Validation hold-out.

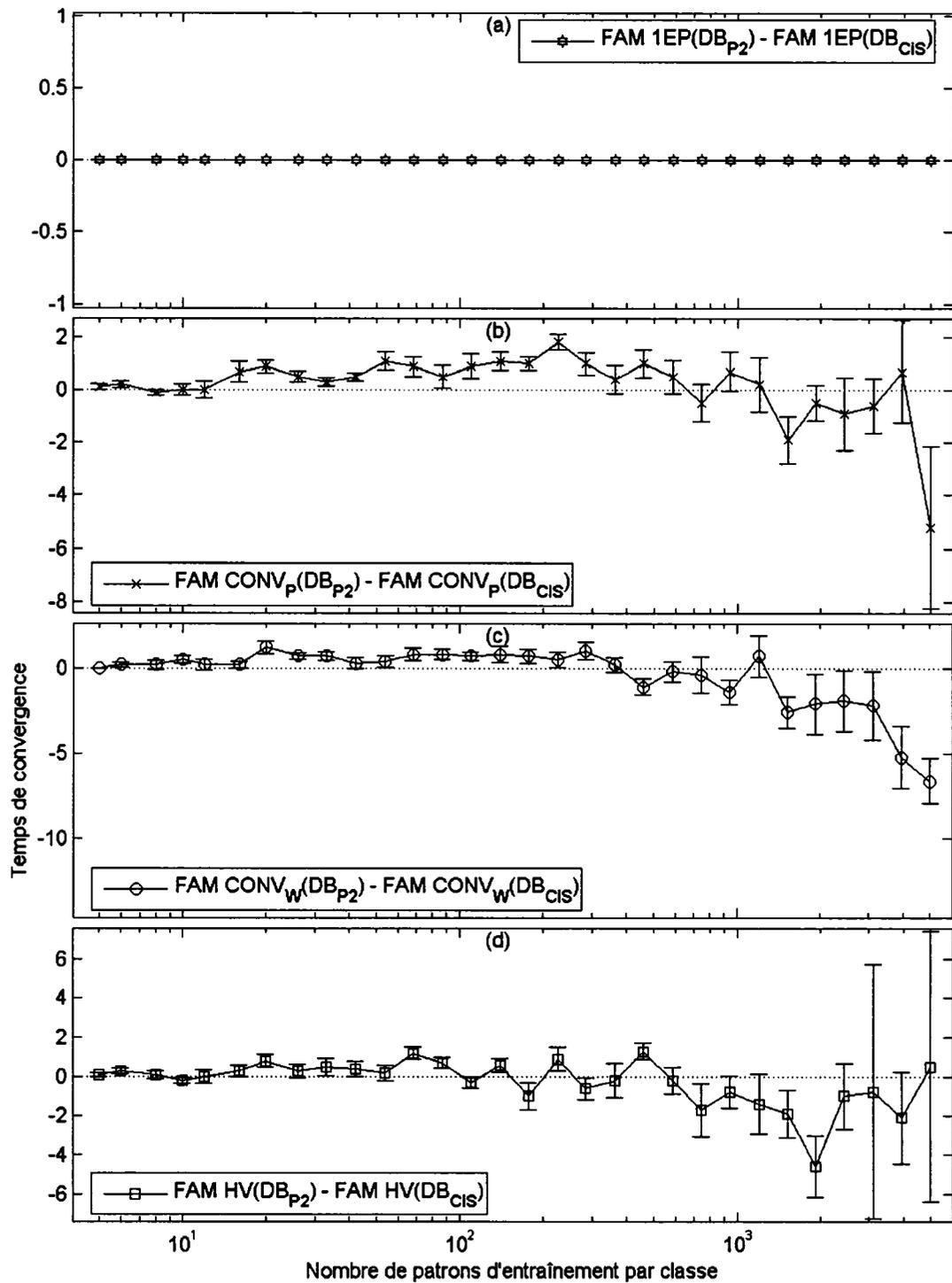


Figure 24 Différence entre DB_{CIS} et DB_{P2} sur le temps de convergence
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

3.2.3 Analyse

En observant les résultats obtenus avec les bases de données avec chevauchement, on ne constate aucune différence majeure entre les deux méthodes de création utilisées. Les erreurs en généralisation, les temps de convergence ainsi que les niveaux de compression sont semblables entre ces deux bases pour un même degré de chevauchement. Aucune des deux structures des bases de données avec chevauchement testées ne montrent un avantage face à l'autre avec les réseaux fuzzy ARTMAP.

En observant les résultats obtenus avec les bases de données sans degré de chevauchement, on constate que les deux bases de données réagissent de la même manière. Ces deux bases ne montrent aucune dégradation des performances due à la taille de la base d'apprentissage et au nombre d'époques. Bien que les courbes présentant les erreurs en généralisation, les temps de convergence ainsi que les niveaux de compression aient les mêmes tendances, il y a de nettes différences au niveau de l'erreur en généralisation et du taux de compression.

Cet effet provient de la complexité des frontières de décision qui jouent un rôle important dans la performance des réseaux fuzzy ARTMAP. Les frontières de décision de la base DB_{P2} sont plus complexes que celles de la base DB_{CIS} . Cette différence engendre pour la base DB_{P2} des erreurs en généralisation plus élevées ainsi que des niveaux de compression plus faibles que ceux obtenus avec la base DB_{CIS} , et ce, pour une même taille de base d'apprentissage.

3.3 Effets de la normalisation

Tel que décrit dans le chapitre 2, deux méthodes de normalisation sont utilisées lors de la réalisation des expérimentations. Ces deux méthodes permettent de comprendre l'impact engendré par la normalisation des données. Les deux méthodes de

normalisation testées sont la normalisation MinMax ainsi que la normalisation Centrée Réduite décrites à la section 2.4.

En étudiant l'impact de ces deux méthodes de normalisation sur les performances en généralisation ainsi que sur la complexité des réseaux créés par le fuzzy ARTMAP nous pourrions favoriser l'une ou l'autre technique de normalisation face à un type de problème de classification donné.

3.3.1 Bases de données avec chevauchement

Cette section présente les différences obtenues entre les résultats des bases ayant un même degré de chevauchement mais dont la technique de normalisation diffère. Les figures 24 à 26 présentent une comparaison entre les techniques de normalisation pour les quatre stratégies d'apprentissage en fonction de la taille de la base d'apprentissage. Étant donné qu'aucune différence n'a été détectée entre les bases DB_{μ} et DB_{σ} , la base DB_{μ} est utilisée.

Le degré de chevauchement de la base DB_{μ} utilisé est de 9%. L'annexe 5 présente ces résultats pour tous les autres degrés de chevauchement.

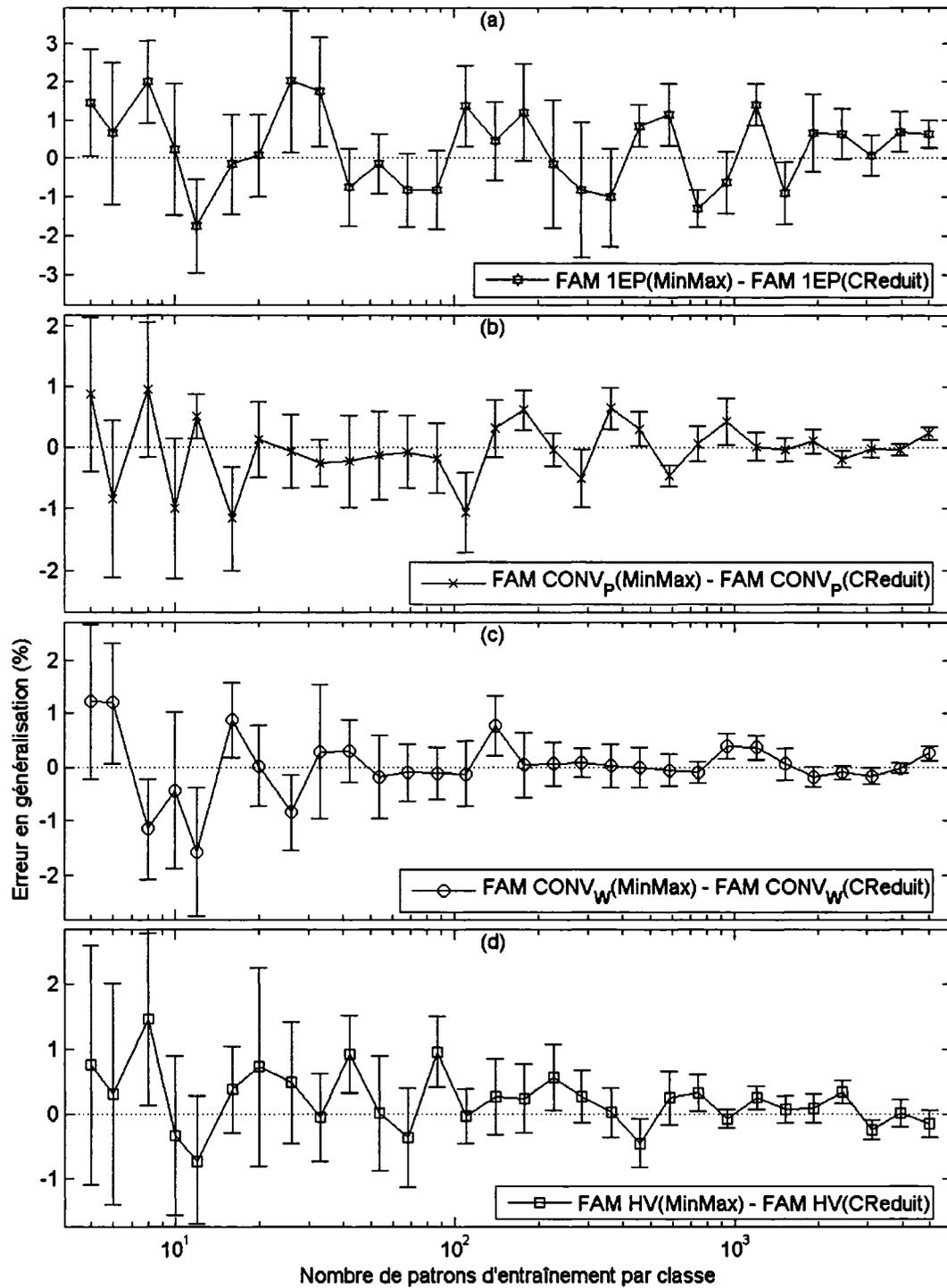


Figure 25 Effet de la normalisation sur l'erreur en généralisation avec $DB_{\mu}(9\%)$
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

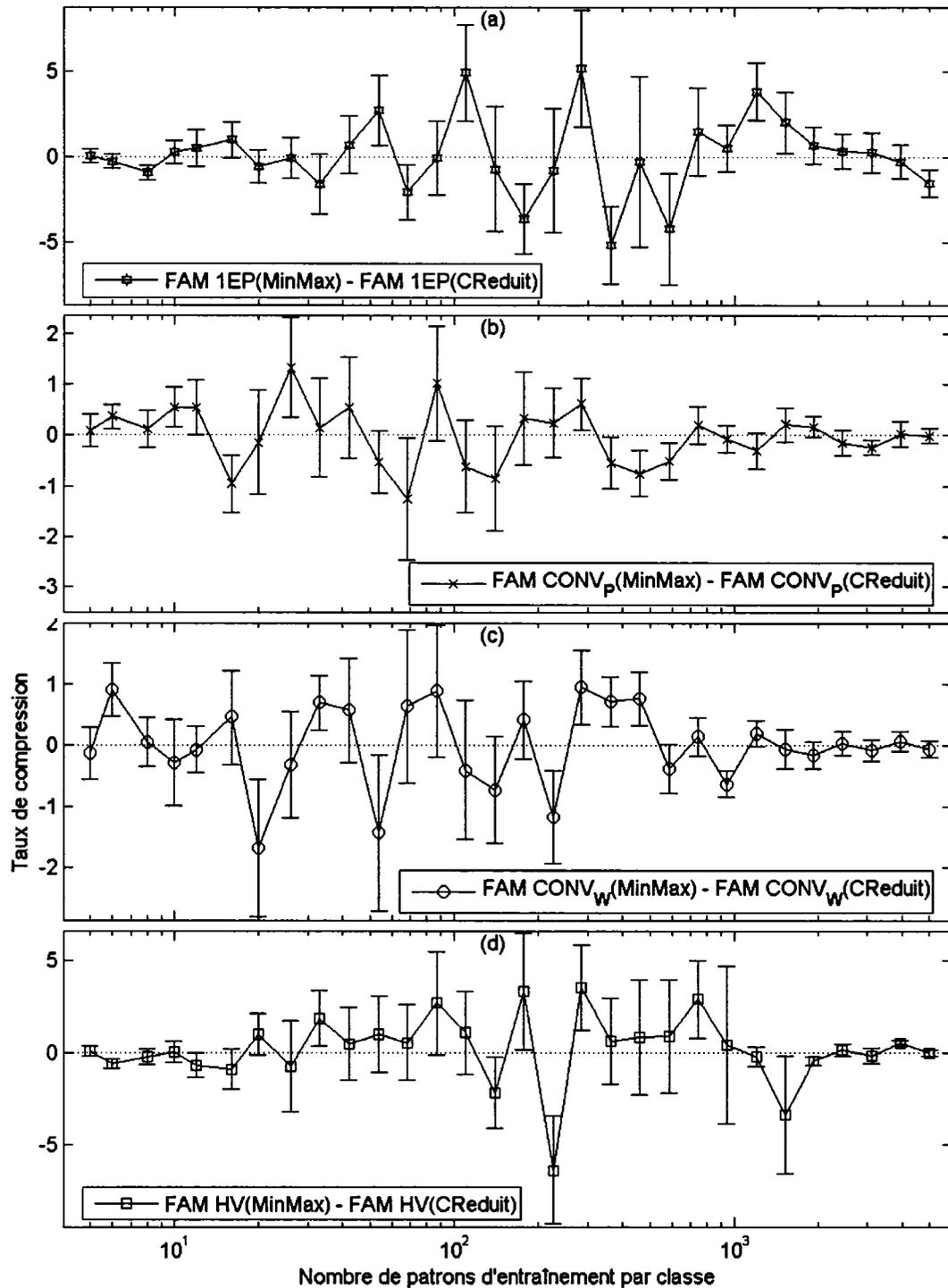


Figure 26 Effet de la normalisation sur le taux de compression avec $DB_{\mu}(9\%)$
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out .

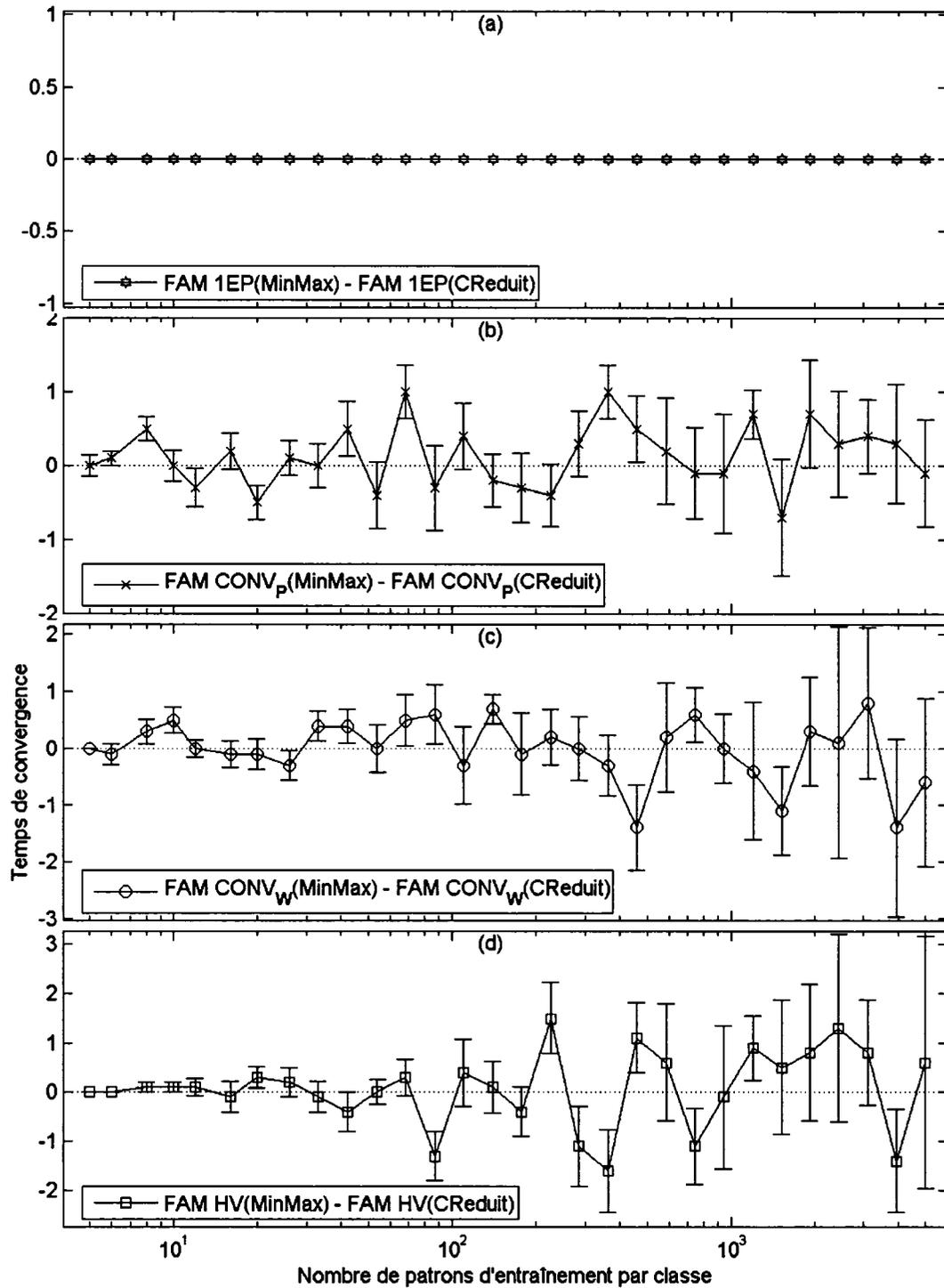


Figure 27 Effet de la normalisation sur le temps de convergence avec $DB_{\mu}(9\%)$
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out

3.3.2 Bases de données sans chevauchement

Les figures 27 à 29 présentent une comparaison entre les deux méthodes de normalisation pour les quatre stratégies d'apprentissage en fonction de la taille de la base d'apprentissage pour la base de données sans chevauchement DB_{CIS} . Cette comparaison présente les erreurs en généralisation, les temps de convergence ainsi que les niveaux de compression. Les résultats obtenus avec la base DB_{P2} sont présentés à la fin de l'annexe 5.

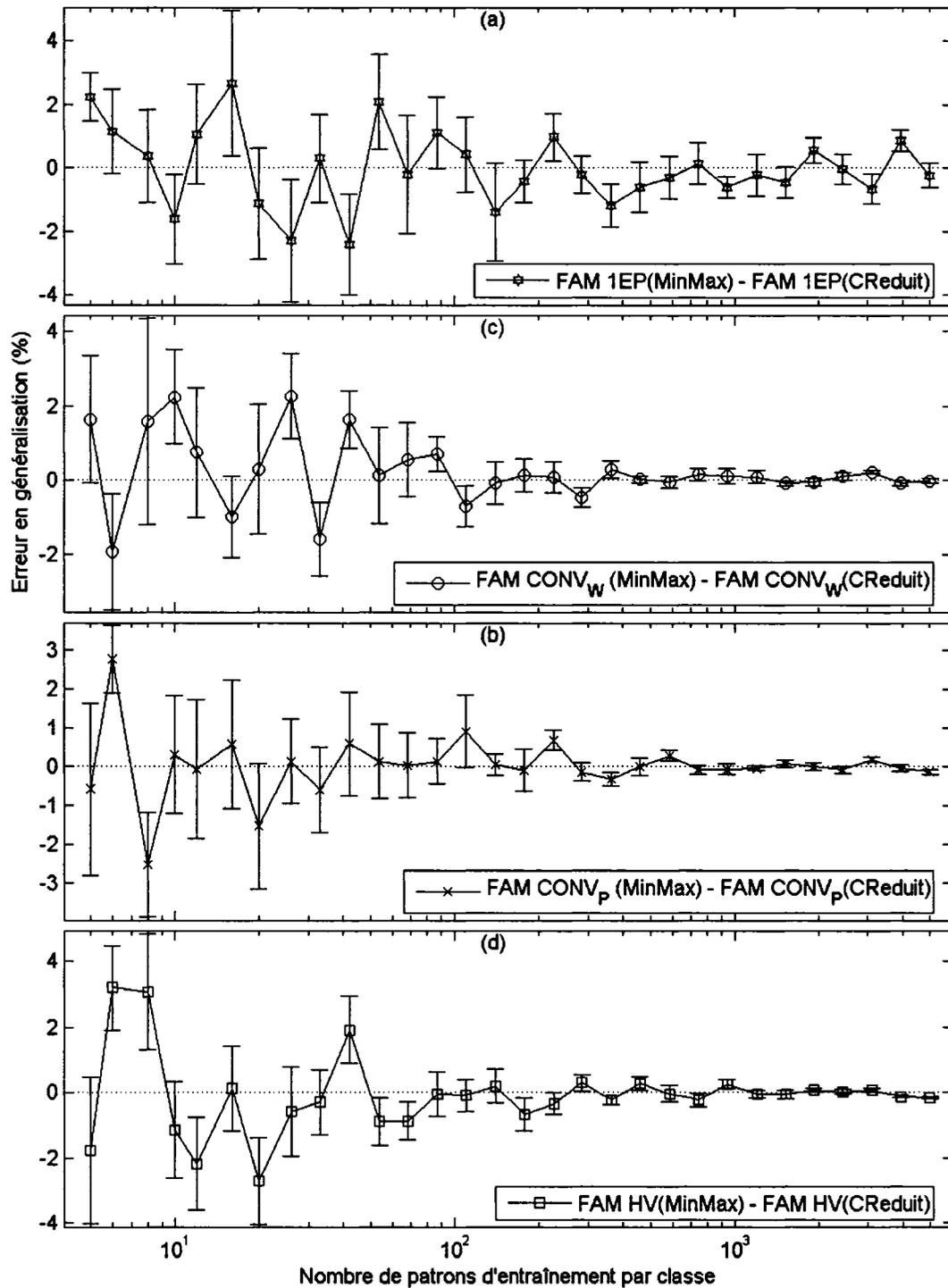


Figure 28 Différence entre l'erreur en généralisation avec la base DB_{Cis}
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

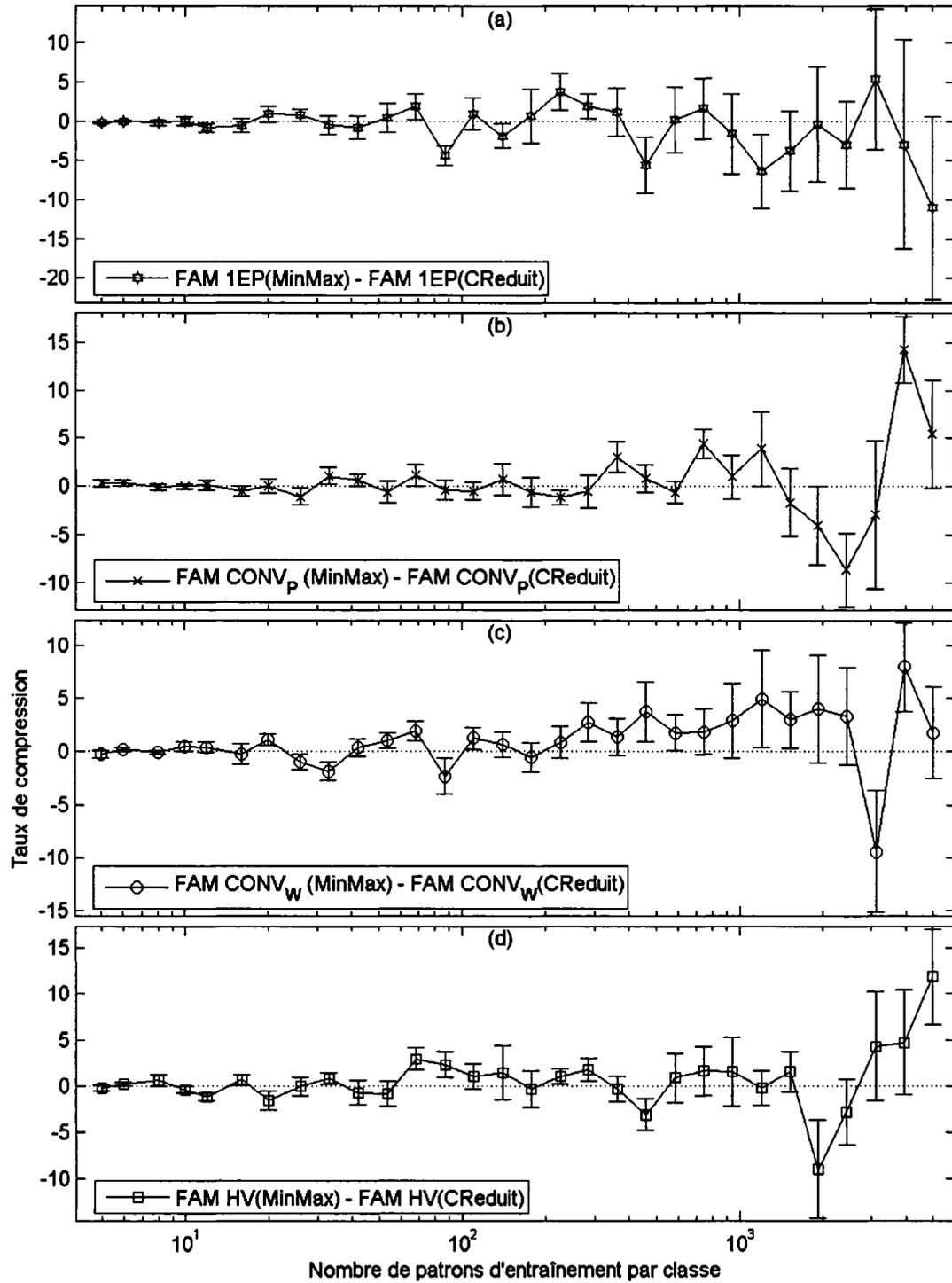


Figure 29 Différence sur le taux de compression avec la base DB_{CIS}

(a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
(d) Validation hold-out.

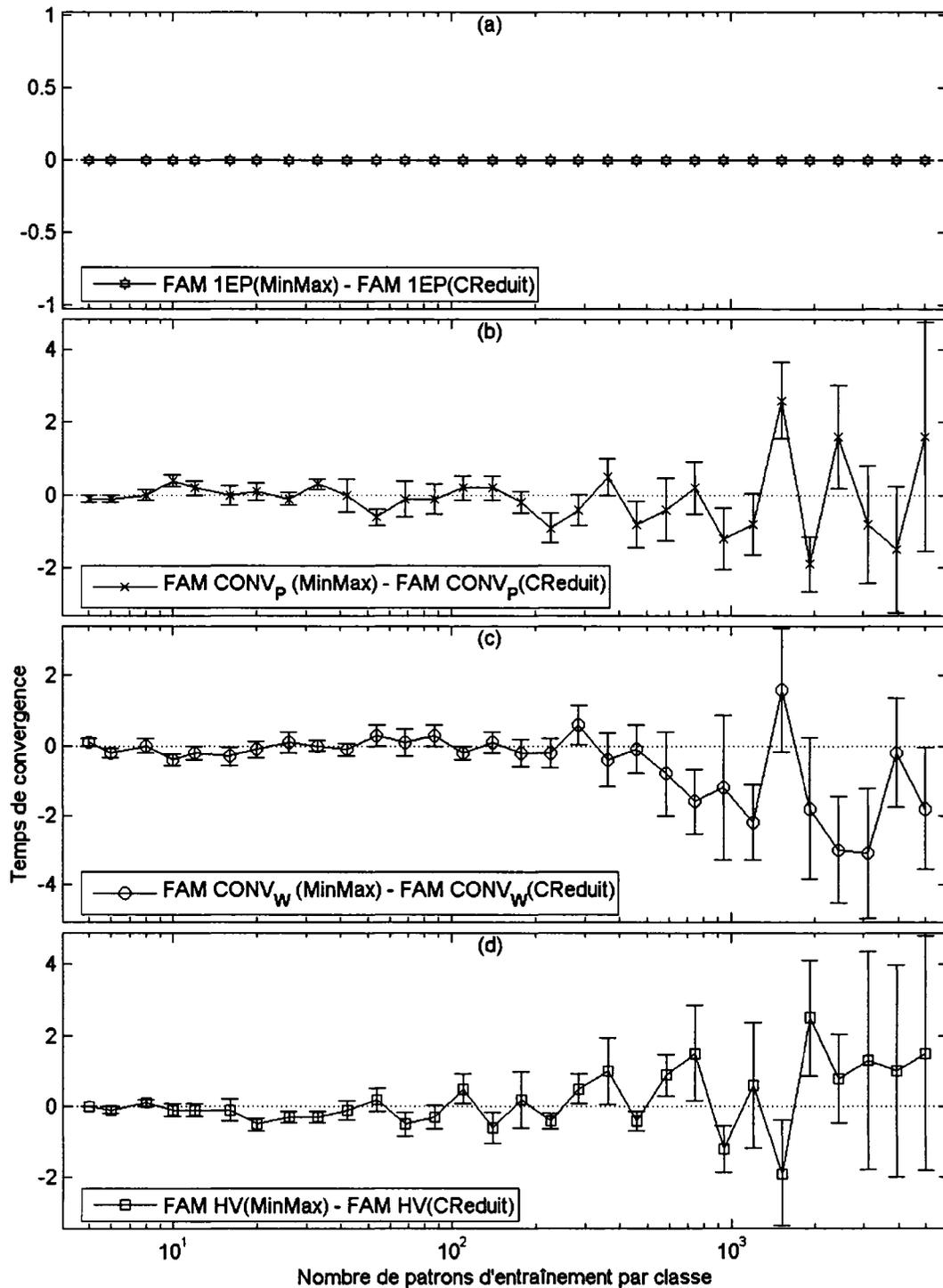


Figure 30 Différence sur le temps de convergence avec la base DB_{Cis}

(a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et (d) Validation hold-out.

3.3.3 Analyse

En observant les résultats obtenus, tant avec les bases de données avec chevauchement qu'avec celles sans chevauchement, on constate qu'il n'y a pas de différences significatives entre les deux méthodes de normalisation utilisées.

En effet, les différences entre les deux techniques de normalisation, tant au niveau de l'erreur en généralisation, du taux de compression, que du temps de convergence, sont minimales et ce, pour l'ensemble des degrés de chevauchement.

Aucune des deux techniques de normalisation n'influence de façon significative les performances en généralisation, le temps de convergence et le niveau de compression des réseaux fuzzy ARTMAP sur les bases de données synthétiques que nous avons testées. A partir d'ici, nous n'utiliserons que la technique de normalisation MinMax.

3.4 Effets de la polarité du MatchTracking

Lors de l'utilisation des réseaux fuzzy ARTMAP, les paramètres internes des réseaux doivent être initialisés. Dans toutes les expériences précédentes, nous avons utilisé les paramètres généraux, admis par la communauté scientifique, soit : $\alpha = 0.01$, $\bar{\rho} = 0.0$, $\beta = 1.0$ et $\varepsilon = 0.001$. Ces valeurs tendent à maximiser les performances et la compression des réseaux fuzzy ARTMAP. Certaines expériences ont démontré qu'en inversant le signe du paramètre epsilon ($\varepsilon = -0.001$), on peut obtenir de meilleurs résultats. L'inversion de ce paramètre est connue sous le nom de MatchTracking négatif (MT-).

Cette section traite de l'influence de la polarité du MatchTracking, soit la différence entre le MatchTracking positif (MT+) et le MatchTracking négatif (MT-). En comparant les résultats obtenus pour les quatre stratégies d'apprentissage, nous sommes en mesure

de déterminer l'influence de la polarité d'épsilon et son impact sur les performances du réseau fuzzy ARTMAP au niveau de l'erreur en généralisation, du temps de convergence ainsi que du taux de compression. Les résultats obtenus dans cette sous-section ont contribué à la publication d'un article [43].

3.4.1 Bases de données avec chevauchement

Cette sous-section présente la comparaison des résultats entre MT- et MT+ pour la base de données $DB_{\mu}(9\%)$ pour l'erreur en généralisation, le taux de compression et le temps de convergence. Ainsi, lorsque la courbe est positive, la valeur obtenue avec MT- est plus grande que celle obtenue avec la base MT+, et vice-versa. Tel que nous pouvons le voir sur les figures suivantes, le fait de changer la polarité du MT du réseau fuzzy ARTMAP influence la performance en généralisation, le temps de convergence ainsi que le taux de compression des réseaux créés. L'annexe 6 présente ces résultats pour tous les autres degrés de chevauchement.

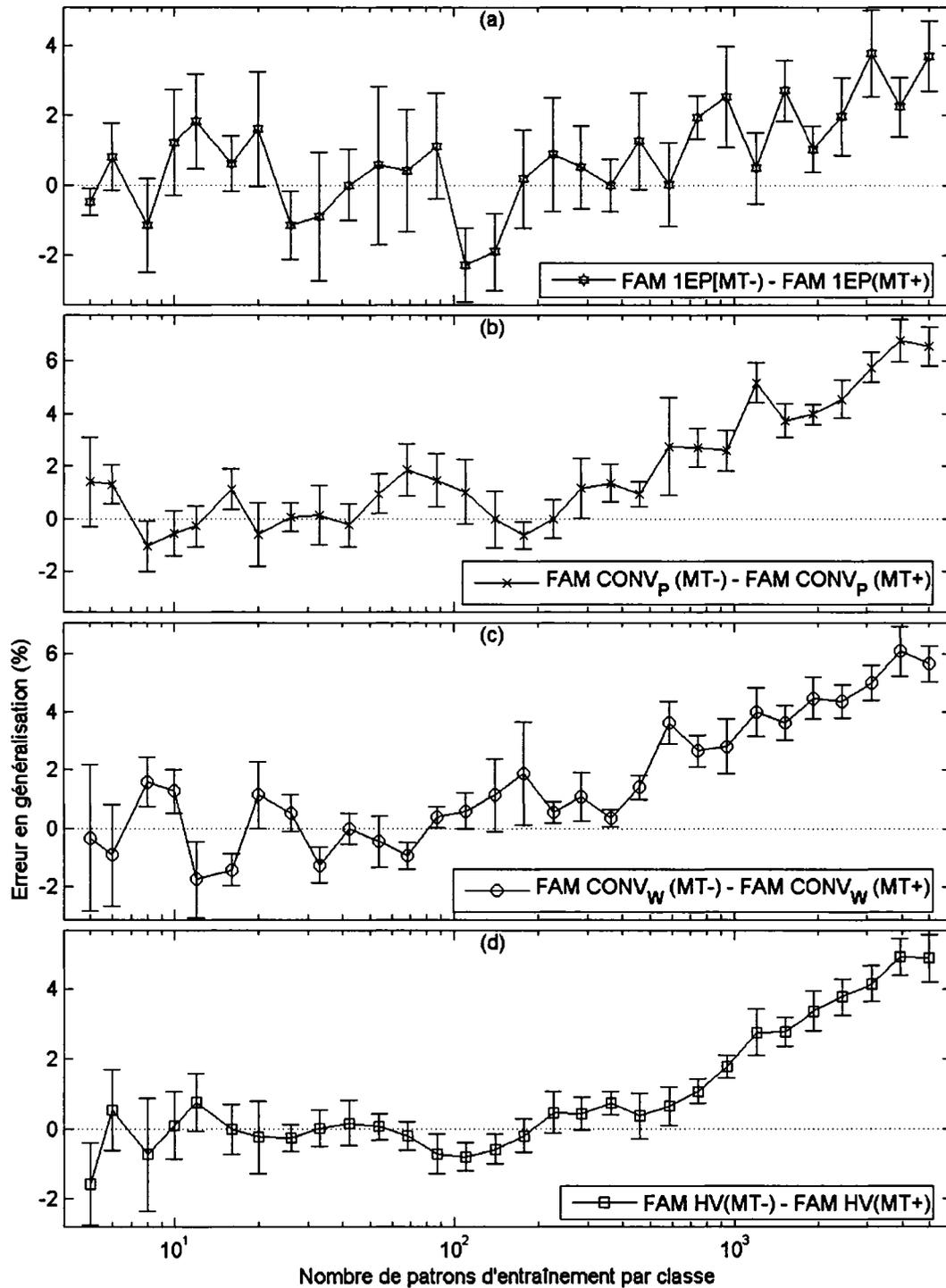


Figure 31 Effet entre MT- et MT+ sur l'erreur en généralisation avec $DB_{\mu}(9\%)$
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

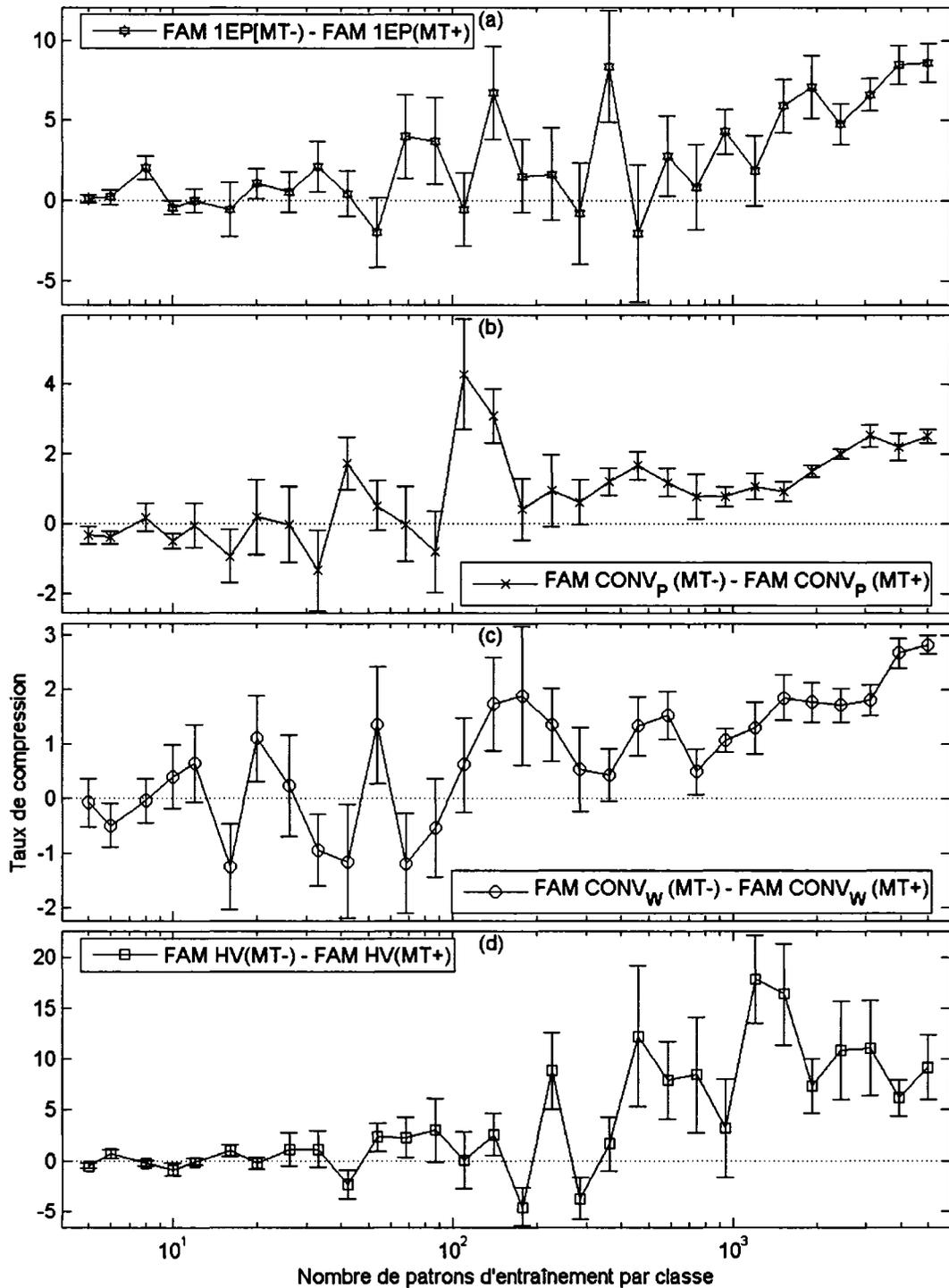


Figure 32 Effet entre MT- et MT+ sur le taux de compression avec $DB_{\mu}(9\%)$
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

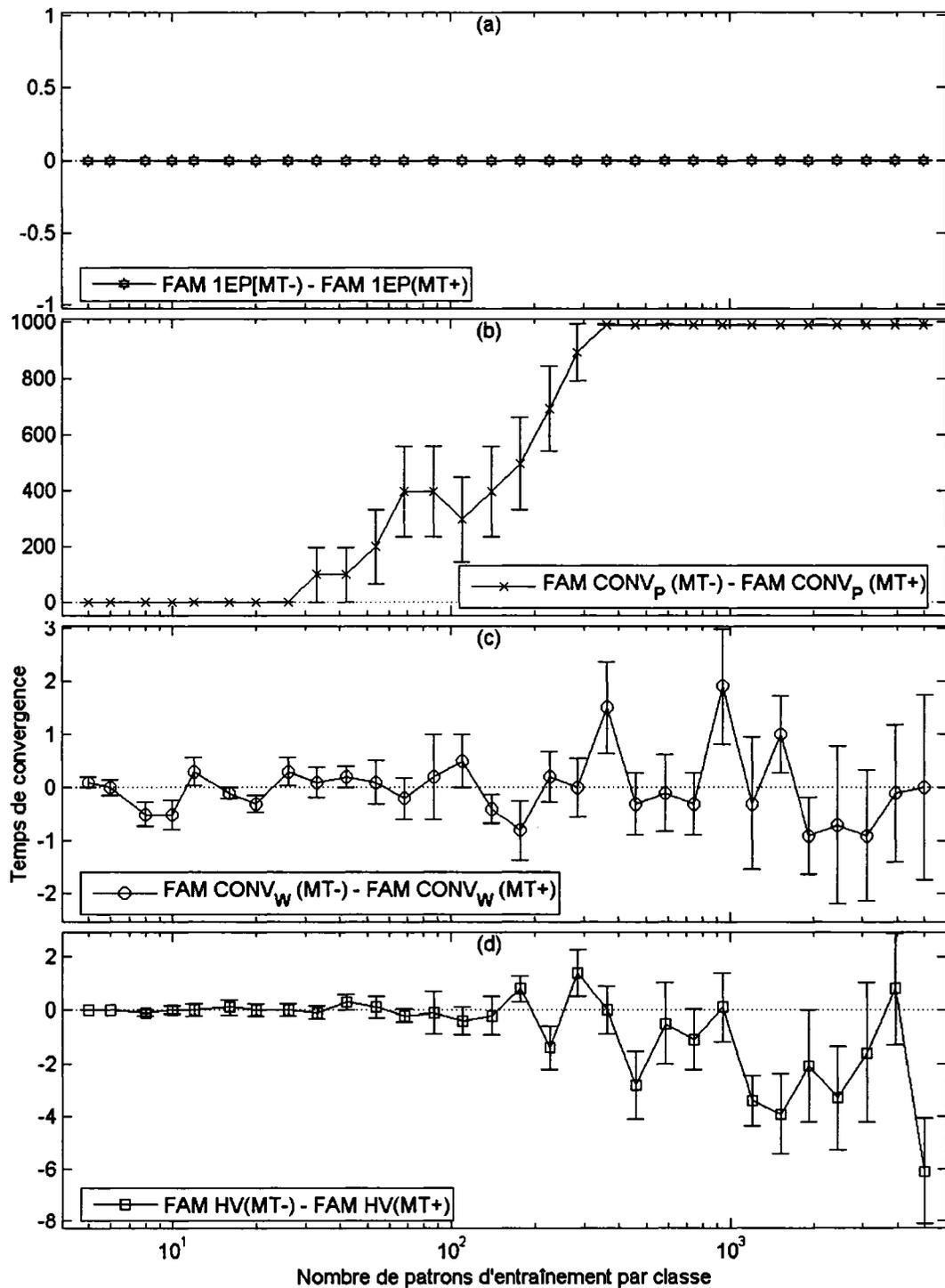


Figure 33 Effet entre MT- et MT+ sur le temps de convergence avec $DB_{\mu}(9\%)$
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

3.4.2 Bases de données sans chevauchement

Cette sous-section présente la comparaison des résultats entre MT- et MT+ pour la base de donnée DB_{CIS} pour l'erreur en généralisation, le taux de compression et le temps de convergence. La technique de normalisation MinMax est utilisée. Les résultats obtenus avec la base DB_{P2} sont présentés à la fin de l'annexe 6.

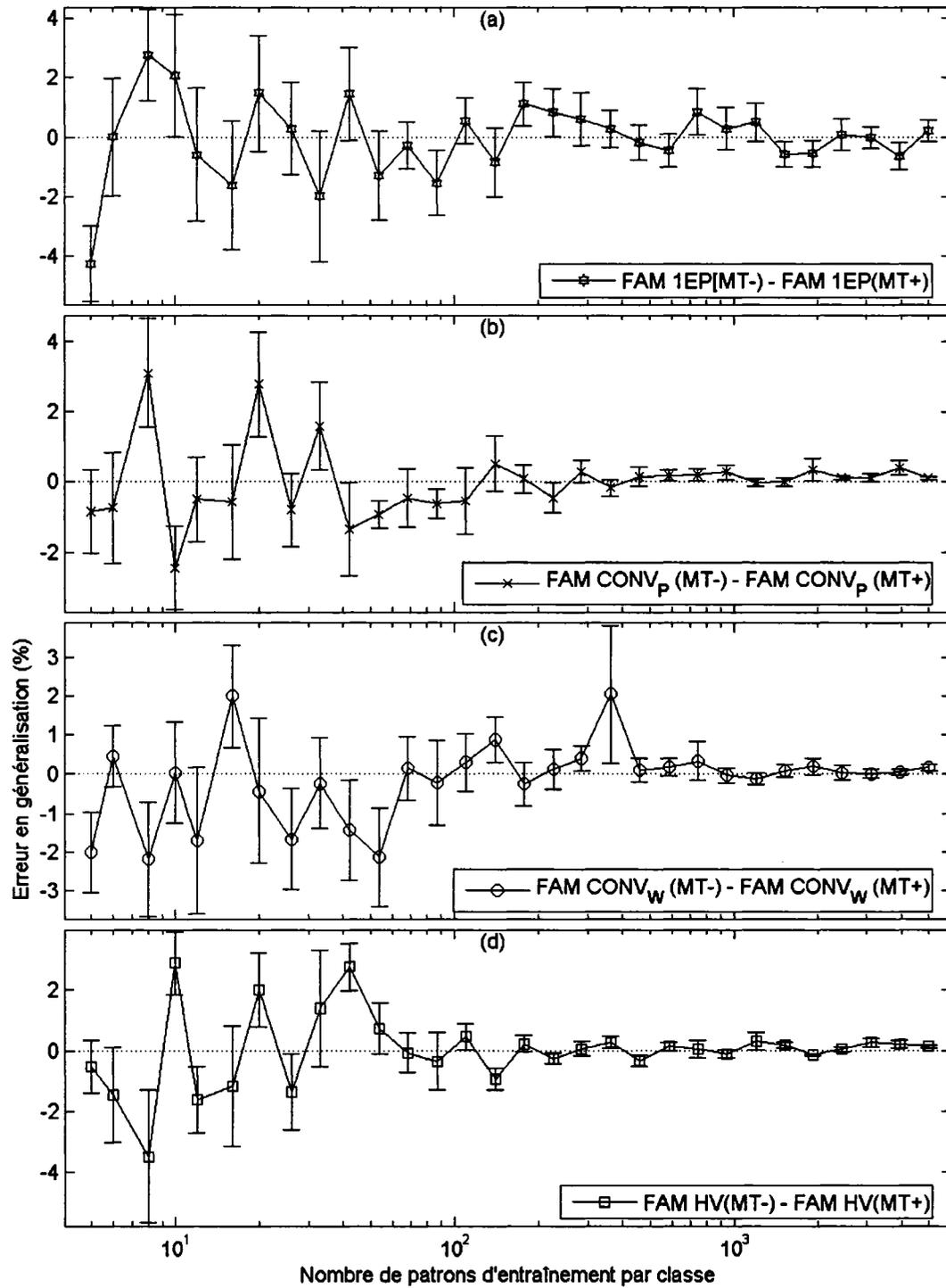


Figure 34 Différence entre MT- et MT+ sur l'erreur en généralisation avec DB_{CIS}
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

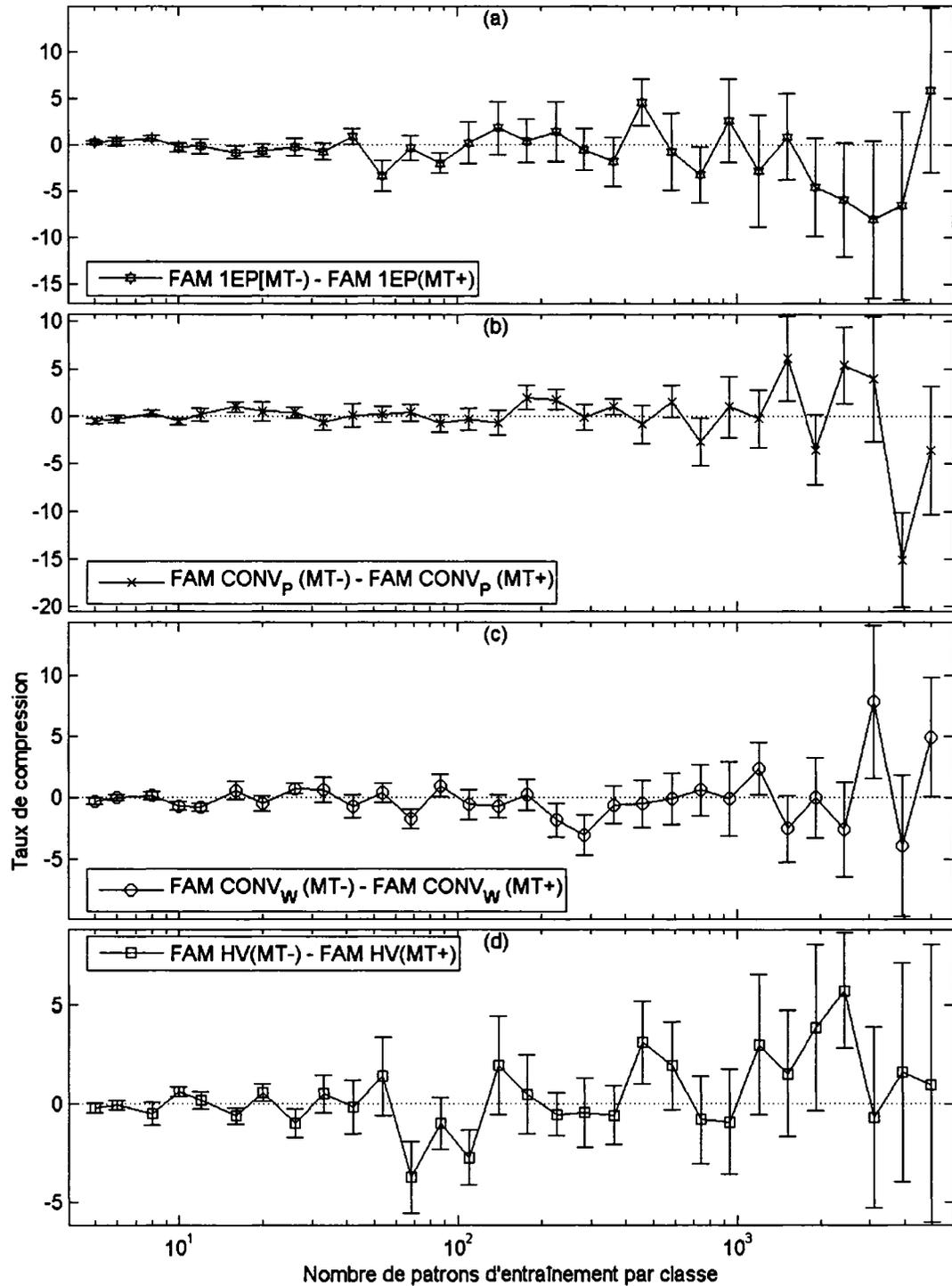


Figure 35 Différence entre MT- et MT+ sur le taux de compression avec DB_{CIS}
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

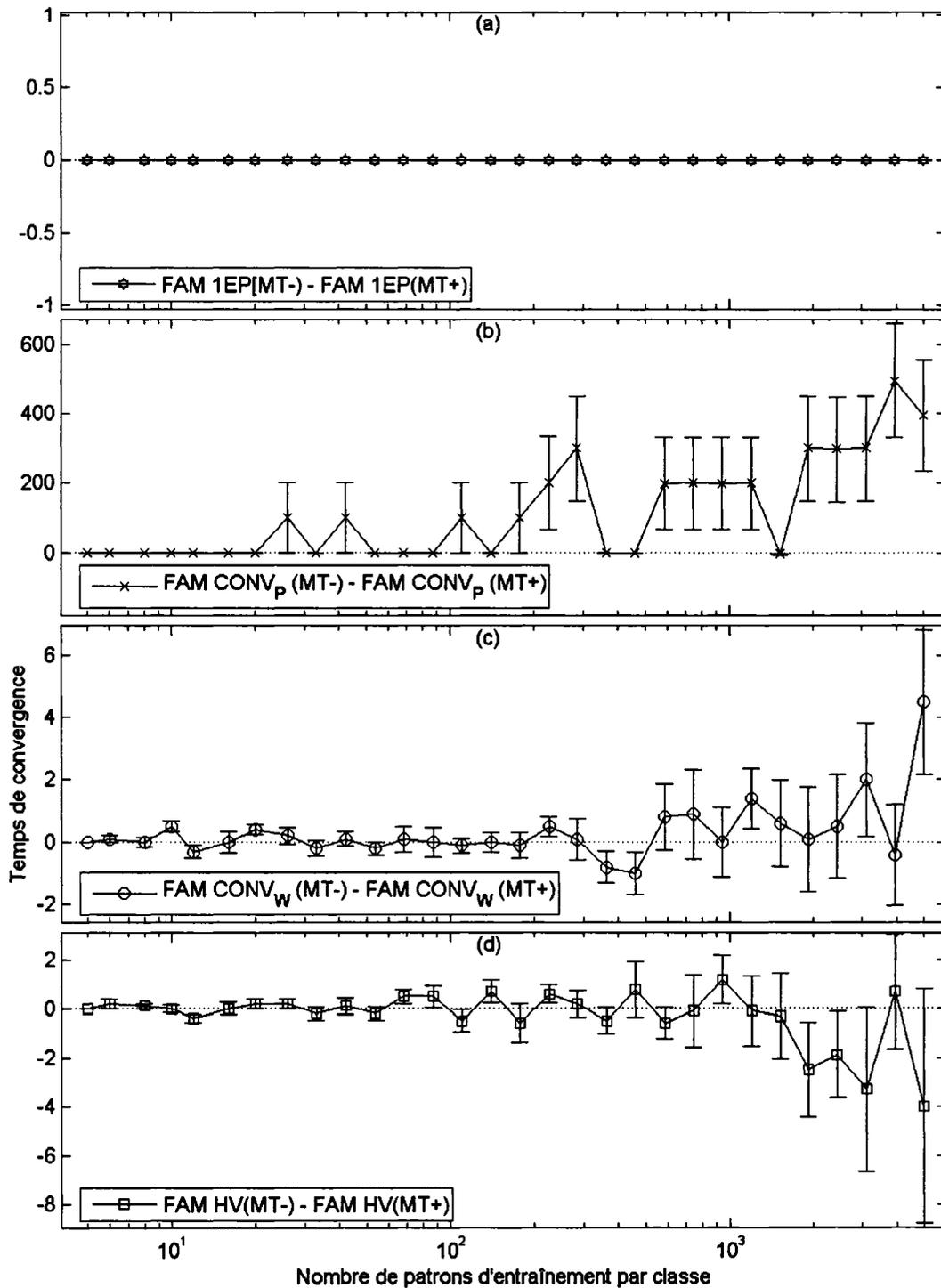


Figure 36 Différence entre MT- et MT+ sur le temps de convergence avec DB_{CIS}
 (a) Une époque, (b) Convergence des patrons, (c) Convergence des poids synaptiques et
 (d) Validation hold-out.

3.4.3 Analyse

Lors de l'utilisation de MT- avec les bases de données avec chevauchement, on remarque que l'erreur en généralisation est plus grande qu'avec MT+ lorsque la taille de la base d'apprentissage augmente. Ainsi, au premier regard, MT+ semble plus approprié pour les bases de données avec chevauchement, car même si MT- obtient de meilleurs taux de compression que MT+, cet avantage est au détriment de la performance en généralisation.

Puisque MT- obtient une plus grande erreur en généralisation avec la taille maximale de la base d'apprentissage, l'erreur de sur-apprentissage engendrée avec ce type de MatchTracking sera plus grande. La figure 37 présente les erreurs de sur-apprentissage pour les quatre méthodes d'entraînement avec MT- ainsi que, comme référence, celles obtenues par la HV avec MT+.

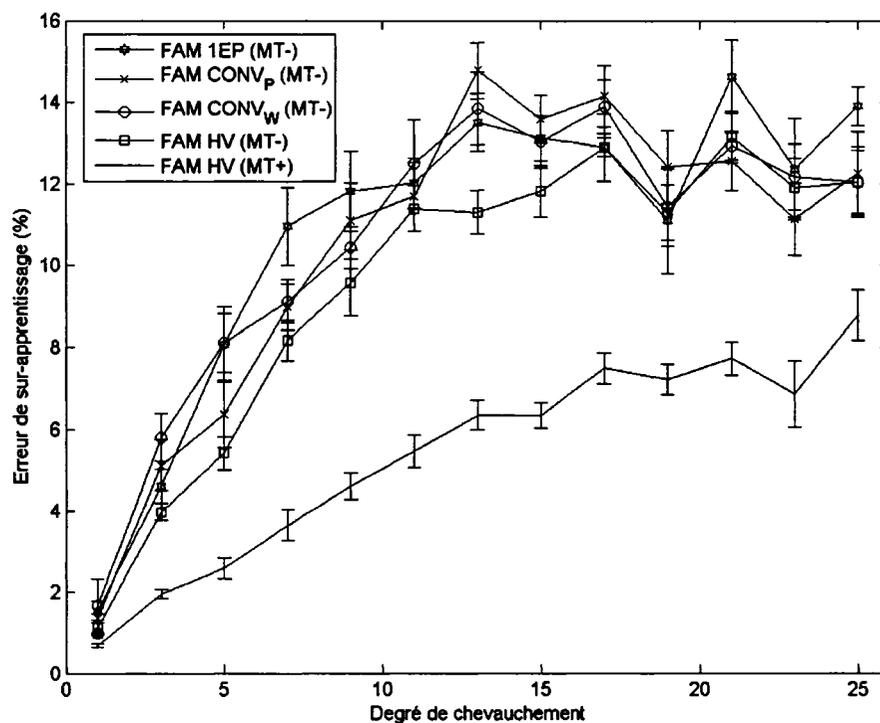


Figure 37 Erreur de sur-apprentissage avec MT-

L'erreur de sur-apprentissage générée par la taille de la base d'apprentissage est bel et bien plus grande avec MT- qu'avec MT+. Mais qu'en est-il de l'erreur nette lorsque la taille de la base d'apprentissage est optimisée? La Figure 38 présente l'erreur nette obtenue avec les bases DB_{μ} pour la stratégie d'apprentissage HV pour les deux types de MatchTracking lorsque la taille de la base d'entraînement est optimisée et lorsque tous les patrons d'entraînement sont utilisés ($5k/\omega$).

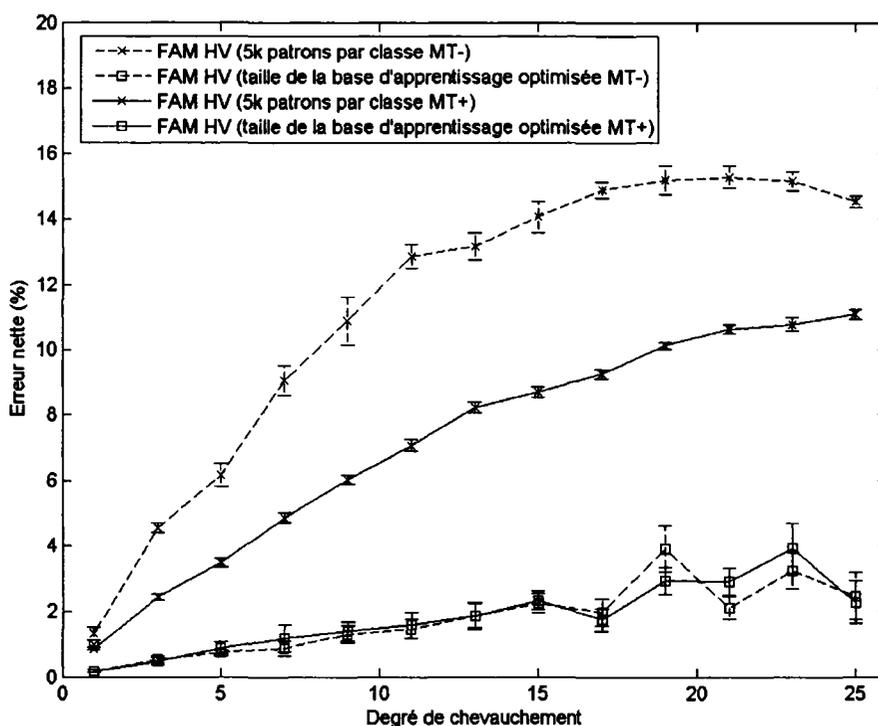


Figure 38 Erreur nette sur DB_{μ} avec la HV pour MT- et MT+

Tel que nous l'avons déjà remarqué avec les figures 30 et 35, l'erreur en généralisation obtenue lors de l'utilisation de la taille maximale de la base d'apprentissage est supérieure lorsque MT- est utilisé. Ceci se reflète à la Figure 38 par l'écart entre les deux courbes lors de l'utilisation de la taille maximale de la base d'entraînement. Par contre, lorsque la taille de la base d'entraînement est optimisée, la différence entre l'erreur nette obtenue avec MT- et MT+ est pratiquement inexistante.

Ainsi, lorsque la taille de la base d'apprentissage est optimisée les deux types de MatchTracking offrent des performances similaires, mais MT- obtient des taux de compression généralement supérieurs comparativement à MT+. Il faut conclure qu'avec l'optimisation de la taille de la base d'entraînement, MT- est plus performant dans le cas des bases de données avec chevauchement.

Si l'on observe les temps de convergence obtenus avec la stratégie d'apprentissage de convergence des patrons (voir figure 33(b)), nous obtenons souvent le nombre maximum d'époques d'entraînement (1000 époques) lors de l'utilisation de MT-. Ceci indique que le réseau ne peut atteindre le premier critère d'arrêt de cette stratégie d'apprentissage, soit une classification parfaite des patrons de la base d'apprentissage. Pourtant lors de l'utilisation de la même base de données avec un MT+, le réseau en était capable. Voyons plus en détails pourquoi ce phénomène survient.

Prenons une situation d'apprentissage avec la stratégie d'apprentissage $CONV_P$ pour 4 patrons de la base d'apprentissage. La Figure 39 présente les catégories créées après une époque d'entraînement. Deux catégories ont été créées.

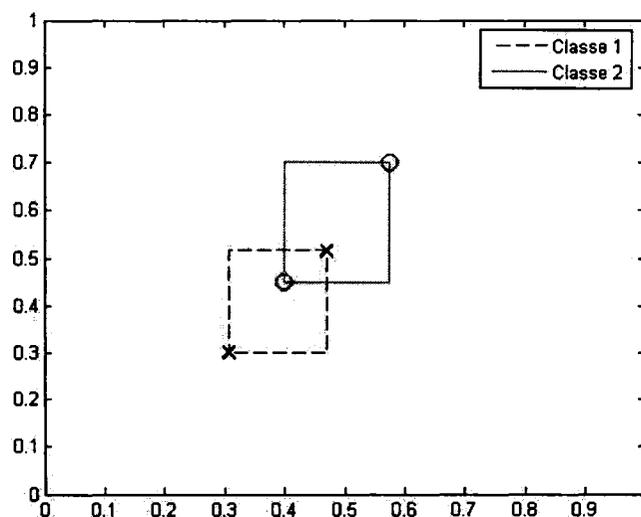


Figure 39 Situation d'apprentissage avec $CONV_P$ pour MT-

La stratégie d'apprentissage par convergence des patrons a, comme condition d'arrêt primaire, la classification parfaite de tous les patrons de la base d'apprentissage. Lorsque le patron de la classe 2 (cercle) qui se trouve dans la catégorie de la classe 1 est testé, le réseau fuzzy ARTMAP cherche, catégorie par catégorie, laquelle représente le mieux ce patron. Puisque ce patron est dans les limites des deux catégories et que la taille de la catégorie appartenant à la classe 1 est légèrement plus petite que l'autre, le réseau sélectionne la catégorie appartenant à la classe 1. Lors de la vérification de la prédiction (phase d'apprentissage), le réseau attribue ce patron à la mauvaise classe. Il élimine donc cette catégorie et regarde si une autre catégorie est capable de classer ce patron avec la même valeur de résonance obtenue par l'ancienne catégorie plus une petite valeur. Cette petite valeur est en fait epsilon. Puisque l'on utilise MT-, la valeur d'epsilon est négative et la catégorie de la classe 2, qui a une taille légèrement supérieure à celle de la classe 1 est sélectionnée et passera le test de la validation de la prédiction.

Ainsi, le réseau ne change pas la valeur des poids synaptiques et ne crée pas de nouvelles catégories. Mais le critère d'arrêt primaire de la stratégie de convergence des patrons ne sera jamais rencontré dans une telle situation et cette stratégie continuera d'essayer d'apprendre jusqu'à l'obtention du nombre maximal d'époques. Ce phénomène ne survient pas lors de l'utilisation de MT+ car, lorsque la catégorie de la classe 2 est rejetée et que le réseau cherche une autre catégorie presque aussi bonne que la première, il n'en trouve aucune et une nouvelle catégorie est créée pour ce point.

De plus, ce phénomène nous permet de juger de la qualité de la stratégie d'apprentissage CONV_p. Bien que cette dernière soit utilisée dans la communauté scientifique, elle n'est pas appropriée pour les réseaux fuzzy ARTMAP. Au lieu d'observer la convergence des patrons, nous devrions regarder la convergence totale des poids synaptiques du réseau. Ainsi, le critère d'arrêt premier de cette stratégie devrait subvenir lorsque deux époques successives obtiennent les mêmes valeurs pour tous les poids synaptiques et non lorsque tous les patrons de la base d'entraînement sont parfaitement classifiés. Le réseau évitera

ainsi d'effectuer des époques d'apprentissage qui ne changent rien aux poids synaptiques du réseau puisque ce dernier a déjà totalement convergé et qu'il ne peut classifier correctement tous les patrons de la base d'entraînement à cause de l'utilisation du MT-.

L'avantage d'utiliser un epsilon négatif (MT-) est de réduire le nombre de catégories créées par le réseau fuzzy ARTMAP, pour ainsi réduire l'effet de la prolifération des catégories et obtenir un meilleur taux de compression. De plus, avec certains types de bases de données, les performances en généralisation sont meilleures avec MT- que lors de l'utilisation de MT+.

Lors de l'utilisation de MT- avec les bases de données sans chevauchement, on remarque que, lorsque la taille de la base d'apprentissage augmente, l'erreur en généralisation obtenue est pratiquement identique à celle obtenue avec MT+. Par contre, lorsqu'on regarde les taux de compression obtenus avec MT- comparativement avec MT+, on remarque qu'il y a beaucoup de variance entre les résultats. Ceci ne nous permet pas de déterminer directement la meilleure polarité du MatchTracking pour les bases de données sans chevauchement. Par contre, nous savons que, généralement, plus le nombre de catégories est élevé lors de l'utilisation des bases sans chevauchement, meilleures sont les performances en généralisation.

Puisqu'il n'y a pratiquement pas de différences entre les erreurs en généralisation obtenues avec MT+ et MT-, on peut en conclure qu'avec les deux types de MatchTracking, aucune dégradation des performances du réseau FAM ne survient avec les bases de données sans chevauchement. Ainsi, plus le nombre de patrons d'entraînement est grand, meilleures sont les performances en généralisation des réseaux fuzzy ARTMAP avec ce type de bases de données. De plus, en examinant le temps de convergence lors de l'utilisation de CONV_P, nous remarquons que, lors de l'utilisation de cette stratégie avec MT-, les temps de convergence correspondent également au nombre maximum d'époques d'apprentissage (1000 époques). Ceci provient du même

phénomène que nous venons d'expliquer pour les bases de données avec chevauchement et qui est présenté à la Figure 39.

3.5 Conclusion

Selon les résultats présentés dans ce chapitre, il est clair que les réseaux fuzzy ARTMAP peuvent subir une dégradation des performances lors de l'utilisation des bases de données ayant du chevauchement. Cette dégradation peut être engendrée par le nombre d'époques d'entraînement ainsi que le nombre de patrons d'entraînement. Lors de la comparaison (voir figure 38) entre l'erreur nette avec le nombre d'époques optimisé (HV) et l'erreur nette avec l'optimisation de ces deux facteurs (HV + nombre de patrons d'entraînement), on peut en déduire qu'entre ces deux facteurs, le nombre de patrons d'entraînement est celui qui dégrade le plus les performances des réseaux fuzzy ARTMAP.

Nous avons également remarqué que MT- est plus approprié pour les bases de données avec degré de chevauchement que MT+ lorsque la taille de la base d'apprentissage est optimisée. Cependant, pour les bases de données sans chevauchement, nous ne sommes pas en mesure de favoriser l'un ou l'autre des deux types de MatchTracking.

De plus, il a été démontré que les deux techniques de normalisation testées obtiennent des résultats équivalents, tout comme les deux structures utilisées dans les bases possédant des degrés de chevauchement.

Finalement, tel que nous l'avons envisagé lors de l'utilisation des bases de données sans chevauchement, la structure de ces bases de données influence les performances des réseaux fuzzy ARTMAP. La base DB_{P2} obtient des résultats plus faibles que la base DB_{CIS} car ses frontières de décision sont plus longues et plus complexes que celles de DB_{CIS} .

Il est à noter que la variance des résultats est parfois grande. Cette variance est majoritairement due au nombre de patrons d'apprentissage. Cependant, l'ordre de présentation des patrons ainsi que la valeur des patrons contenue dans les dix différentes bases générées pour chaque cas influencent également la variance des résultats.

Le tableau IX présente un sommaire des résultats obtenus lors de l'utilisation de la taille maximale de la base d'apprentissage avec les quatre stratégies d'entraînement pour l'ensemble des bases de données. Ces résultats sont l'erreur en généralisation moyenne et la dispersion des résultats sur les 10 répliques. L'annexe 7 présente une synthèse des résultats pour toutes les bases de données.

Tableau IX

Résultats sommaires avec 5k patrons par classe

Stratégies d'apprentissage	Erreur en généralisation moyenne (dispersion des résultats) %				
	DB _μ (1%)	DB _μ (9%)	DB _μ (25%)	DB _{CIS}	DB _{P2}
Erreur théorique	1,00	9,00	25,00	0,00	0,00
CQB	1,00 (0,04)	9,12 (0,08)	25,11 (0,10)	ND	ND
kNN	1,08 (0,03)	9,88 (0,08)	27,23 (0,12)	0,86 (0,03)	1,65 (0,04)
1NN	1,54 (0,03)	13,35 (0,10)	33,49 (0,16)	0,84 (0,02)	1,61 (0,04)
FAM 1EP MT-	2,75 (0,20)	22,49 (0,87)	40,58 (0,47)	4,20 (0,25)	8,89 (0,44)
FAM HV MT-	2,17 (0,08)	20,80 (0,56)	39,83 (0,31)	1,69 (0,07)	4,26 (0,16)
FAM CONV _w MT-	2,74 (0,18)	20,45 (0,46)	40,31 (0,27)	1,77 (0,09)	4,48 (0,38)
FAM CONV _p MT-	2,56 (0,09)	22,00 (0,66)	40,42 (0,31)	1,59 (0,04)	4,51 (0,19)
FAM 1EP MT+	2,51 (0,14)	18,78 (0,38)	38,81 (0,36)	3,98 (0,21)	7,33 (0,33)
FAM HV MT+	1,88 (0,05)	15,17 (0,13)	36,10 (0,20)	1,58 (0,05)	3,68 (0,07)
FAM CONV _w MT+	1,97 (0,09)	15,30 (0,16)	35,94 (0,15)	1,64 (0,05)	3,66 (0,08)
FAM CONV _p MT+	1,90 (0,07)	15,44 (0,15)	36,14 (0,20)	1,47 (0,05)	3,61 (0,08)