Titre: Title:	Efforts de calcul et débordements de décodeurs séquentiels à pile
Auteurs: Authors:	David Haccoun
Date:	1985
Type:	Rapport / Report
Référence: Citation:	Haccoun, D. (1985). Efforts de calcul et débordements de décodeurs séquentiels à pile. (Technical Report n° EPM-RT-85-15). https://publications.polymtl.ca/9633/

Institution:	École Polytechnique de Montréal
Numéro de rapport: Report number:	EPM-RT-85-15
URL officiel: Official URL:	
Mention légale: Legal notice:	

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
SECTION COMMUNICATION, INFORMATIQUE

EPM/RT-85-15

EFFORTS DE CALCUL ET DEBORDEMENTS DE DECODEURS SEQUENTIELS A PILE

par

David HACCOUN, Ing., Ph.D.
Professeur titulaire
Département de génie électrique
Ecole Polytechnique de Montréal
Montréal, Québec, Canada

MAI 1985

Ecole Polytechnique de Montréal

Campus de l'Université de Montréal
Case postale 6079
Succursale 'A'
Montréal, Québec
H3C 3A7

ÉCOLE POLYTECHNIQUE DE MONTRÉAL
SERVICE DE L'ÉDITION

Nº de série
EPM/RT- 85-15

PUBLICATION ET DIFFUSION DES RAPPORTS TECHNIQUES
autorisations et renseignements

TITRE: EFFORTS DE CALCUL ET DEBORDEMENTS DE DECODEURS SEQUENTIELS A PILE*

DATE DE PUBLICATION: 10-05-85 1er TIRAGE: 40 copies

AUTORISATION DE PUBLIER

Par la présente, j'autorise DAVID HACCOUN
(nom(s) auteur(s))

à publier le rapport technique dont le titre apparaît ci-dessus.

J. J. Hauke
(signature directeur) (entité administrative)

AUTORISATION DE DIFFUSER

Par la présente, j'autorise l'École Polytechnique de Montréal à reproduire le rapport technique dont le titre apparaît ci-dessus et à le vendre aux personnes et aux organismes qui en feront la demande, jusqu'à ce que ce que j'aie signifié par écrit au Service de l'édition ma décision d'en arrêter la diffusion. Je me réserve tous les autres droits de publication.

(signature(s) auteur(s))

RENSEIGNEMENTS POUR LE DÉPÔT LÉGAL

	.Nom et prénom de l'auteur	.Date de naissance a - m - j	.L'auteur est-il: né au Canada? oui	non	citoyen canadien? oui	non	résidant du Canada? oui	non
1.	David Haccoun	37 07 04	☐	☒	☒	☐	☒	☐
2.			☐	☐	☐	☐	☐	☐
3.			☐	☐	☐	☐	☐	☐
4.			☐	☐	☐	☐	☐	☐

EFFORTS DE CALCUL ET DÉBORDEMENTS DE DÉCODEURS SÉQUENTIELS À PILE*

David HACCOUN

Professeur titulaire
Département de génie électrique
École Polytechnique de Montréal
Montréal, Québec, Canada

RÉSUMÉ

Cet article traite de codage convolutionnel et de décodage séquentiel par l'algorithme à pile de Zigangirov-Jelinek et de certaines de ses variantes. Ces variantes ont toutes pour objectif la diminution de la variabilité de l'effort de calcul du décodage séquentiel. Utilisant la simulation sur ordinateur, on montre que cette variabilité peut être grandement diminuée au coût d'un accroissement de l'effort de calcul moyen, mais aussi sans dégradation de la performance d'erreur.

Le remplissage de la pile du décodeur est examiné ainsi que son impact sur la taille de la file d'attente dans le tampon d'entrée du décodeur. Une analyse simple de la file d'attente a montré qu'en choisissant judicieusement le gain de vitesse du décodeur, la taille de la pile et la longueur des blocs, on peut limiter la taille du tampon d'entrée à deux longueurs de bloc sans risque de débordements. Enfin on montre qu'en controllant les débordements de la pile et en utilisant une procédure de retransmission des blocs ayant fait déborder la pile, on peut controller la probabilité de débordement du tampon d'entrée et réduire sensiblement la probabilité d'erreur. Ces avantages sont obtenus au coût d'une faible réduction du taux de codage effectif et d'une légère augmentation de la complexité du décodeur.

* Cette recherche a été supportée en partie par une subvention du Conseil de recherches en sciences naturelles et en génie du Canada.

EFFORTS DE CALCUL ET DÉBORDEMENTS DE DÉCODEURS SÉQUENTIELS À PILE

David Haccoun

1. INTRODUCTION

L'usage de plus en plus répandu de techniques de transmissions numériques dans les télécommunications terrestres et par satellite conduit à l'utilisation grandissante de procédures de correction d'erreur par codage de canal qui sont puissantes, fiables et pratiques. Aussi un problème important consiste à développer des techniques de codage et décodage délivrant de faibles probabilités d'erreur avec des décodeurs de complexité acceptable. Dans les canaux de communication sans mémoire, les systèmes utilisant le codage convolutionnel avec décodage probabiliste sont parmi les plus intéressants tant du point de vue de leur performance d'erreur que du point de vue de leur réalisation et implantation matérielle. Le décodage probabiliste comprend un ensemble de techniques où le message décodé est obtenu par des procédures probabilistes plutôt que par des opérations algébriques fixes, et où les codes utilisés n'ont pas, en principe, à satisfaire à une structure algébrique particulière comme pour les codes en blocs. Ces codes peuvent être choisis au hasard sans nuire à la technique de décodage, ce qui permet d'augmenter considérablement leur champ d'application.

Les deux principales techniques de décodage probabiliste des codes convolutionnels sont le décodage séquentiel [1] et le décodage de Viterbi [2]. Chacune de ces techniques consiste à trouver un chemin particulier (le message transmis) dans un graphe orienté (arbre ou treillis), où on assigne aux branches des valeurs de vraissemblance γ_j (appelées "métriques") entre les symboles reçus du canal de transmission et les symboles codés qui

auraient pu être transmis. L'objectif général du décodeur est donc de déterminer le chemin ayant la métrique totale $\Gamma = \sum_j \gamma_j$ la plus élevée, et ce, avec un minimum d'effort et un maximum de fiabilité. Ce chemin de métrique maximum trouvé par le décodeur est la séquence décodée $\underline{X}$, de laquelle on déduit la séquence d'information $\underline{U}$ qui est alors transférée à l'utilisateur. Le décodeur commet une erreur de séquence non détectée si $\underline{U} \neq \underline{U}$, où $\underline{U}$ est la séquence d'information transmise par la source.

Les techniques de décodage séquentiel et de décodage de Viterbi sont nettement différentes l'une de l'autre, ont des performances d'erreur et des domaines d'application différents ce qui les distinguent l'une de l'autre en plus de les distinguer des techniques de codage en bloc [3].

La Figure 1 montre les courbes de performance et les gains de codage de plusieurs systèmes de codage utilisant une modulation de type PSK cohérente parfaite [3]. Le gain de codage d'un système de codage est égal à la différence en dB des valeurs de E_b/N_o requises pour une probabilité d'erreur donnée entre ce système de codage et la modulation PSK cohérente parfaite sans codage. Ici, E_b est l'énergie reçue par bit d'information, N_o la densité spectrale du bruit, et le rapport signal-à-bruit par bit d'information E_b/N_o sert de facteur de mérite pour comparer les performances de différents systèmes de modulation et de codage. Le problème de base des systèmes codés peut se résumer à déterminer le système qui fournira une performance d'erreur donnée avec la plus faible valeur de E_b/N_o. Se référant à la Figure 1, par exemple, au taux d'erreur de 10^{-5}, le codage en bloc BCH (128,112) donne un gain de codage de 2 dB, alors que le décodage de Viterbi avec quantification pondérée ($K = 7$, $R = \frac{1}{2}$) permet un gain de codage égal à 5.0 dB. On peut voir qu'un décodeur séquentiel avec quantification ferme permet un gain de codage égal à 5.2 dB, alors qu'en quantification pondérée à 8 niveaux ou 3 bits, le gain de codage peut atteindre 7 à 8 dB. D'un point de vue pratique, un gain de 5.2 dB peut se traduire soit par une

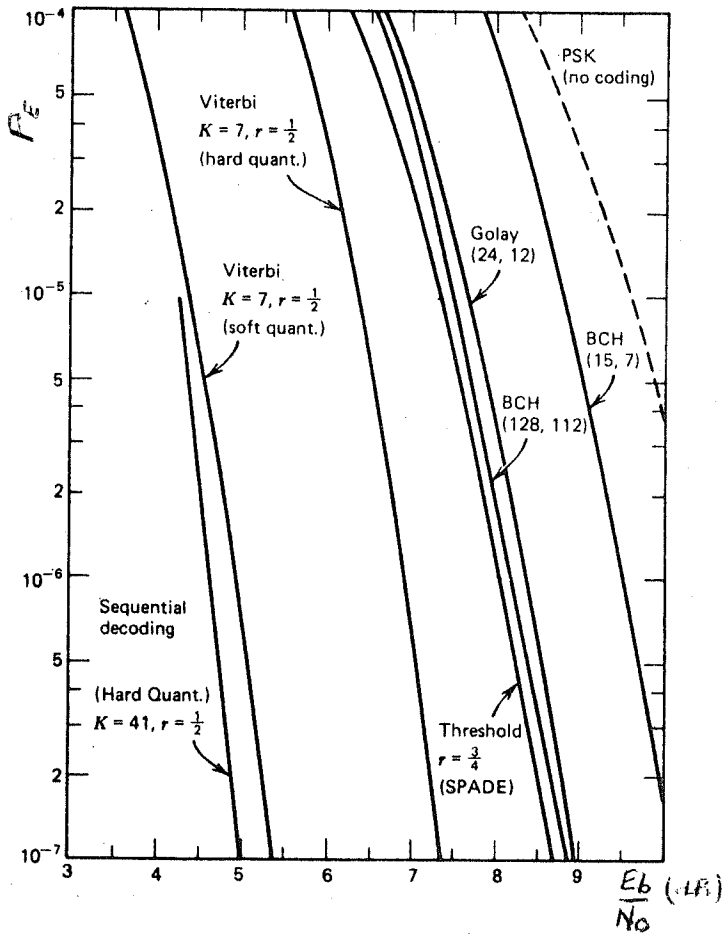

Figure 1: Courbes de performance de plusieurs systèmes codés [3]

réduction de 5.2 dB de la puissance de transmission de l'émetteur, soit par une augmentation de la vitesse de transmission des données non codées par un facteur de $10^{0.52} = 3.3$. Dépendant des applications, chacune de ces alternatives peut s'avérer particulièrement intéressante pour améliorer le design d'un système, en particulier dans les liaisons numériques par satellite où chaque décibel d'énergie transmise par le satellite est extrêmement coûteux.

Cet article traite essentiellement de codage covolutionnel et de décodage séquentiel en particulier de l'algorithme de Zigangirov - Jelinek (algorithme à pile) et de certaines de ses variantes. Après avoir brièvement rappelé la structure des codes convolutionnels et présenté l'algorithme à pile de base, quelques variantes du décodage séquentiel à pile qui permettent de diminuer la variabilité de l'effort de calcul sont présentées. Le comportement du remplissage de la pile est examiné et l'impact sur la dynamique de la file d'attente au tampon d'entrée est analysé. Enfin on présente une procédure d'utilisation de décodeurs séquentiels à pile où les blocs difficiles à décoder provoquent un débordement de la pile et sont retransmis. Cette procédure qui fait du décodeur séquentiel un décodeur hybride détecteur - correcteur d'erreur permet un échange de la mémoire et du gain de vitesse du décodeur. En particulier on montre qu'avec un choix judicieux de la taille de la pile et du gain de vitesse, un tampon de taille deux longueurs de bloc ne débordera jamais.

2. CODAGE CONVOLUTIONNEL ET DÉCODAGE SÉQUENTIEL

Structure des codes convolutionnel

Un codeur convolutionnel de taux de codage R = 1/V peut être représenté par une machine linéaire à états finis composée d'un registre à décalage de K cellules, de V additionneurs modulo-2 connectés à certaines cellules du registre à décalage, et d'un commutateur qui balaye les V additionneurs modulo-2. L'ensemble des connexions entre le registre à décalage et les additionneurs modulo-2 spécifie le code. Par exemple, un codeur convolutionnel K = 3, R = ½ est montré à la Figure 2.

Un codeur convolutionnel fonctionne comme suit: les bits d'information sont introduits par la gauche, un bit à la fois, et après chaque décalage, les additionneurs modulo-2 sont échantillonnés en séquence par le commutateur, fournissant ainsi V symboles codés qui sont modulés et transmis dans le canal. Le taux de codage est donc R = 1/V. Pour ces codeurs binaires simples, la longueur K du registre à décalage s'appelle la longueur de contrainte du code. Un codeur peut être facilement généralisé, et admettre non pas 1 mais n bits à la fois dans le codeur, avec $n \leq V$, et le taux de codage devient alors R = n/V.

Arbre et treillis

Considérant seulement des codes convolutionnels de taux R = 1/V et de longueur de contrainte K, à chaque bit d'information il y correspond 2 branches d'un arbre portant chacune V symboles codés. L'extrêmité de chaque branche est un noeud caractérisé par un état du codeur. L'état du codeur est le contenu des (K-1) premières cellules du registre à décalage, et donc le nombre d'états distincts est égal à $2^{(K-1)}$.

Un chemin dans l'arbre est spécifié par la séquence d'information qui est entrée dans le codeur et deux chemins reconvergent (i.e. ont le même état terminal) si leurs (K-1) derniers bits d'information sont identiques [3]. Au-delà d'une profondeur égale à (K-1) l'arbre d'encodage contient donc une énorme redondance qui peut être éliminée en ne gardant qu'un seul chemin au-delà de chaque noeud de reconvergence. L'arbre devient alors un treillis ayant 2^{K-1} états, et pour une séquence d'information de longueur L bits, les chemins dans l'arbre ou le treillis ont donc une longueur maximale égale à L branches. Un exemple d'arbre et de treillis correspondant au codeur de la Figure 2 est donné aux Figures 3 et 4 respectivement.

Les notions de chemin, arbre et treillis sont essentielles à la compréhension du codage et décodage des codes convolutionnels. La séquence d'information étant représentée par un chemin (le chemin correct), la fonction de décodage consiste donc, connaissant la séquence reçue, à trouver le chemin dans l'arbre ou le treillis qui soit le plus "vraisemblable", c.a.d. qui "ressemble" le plus à la séquence reçue. Le décodage séquentiel dont il est question dans le reste de cet article est une des techniques parmi les plus puissantes et les plus efficaces pour trouver ce chemin le plus vraisemblable.

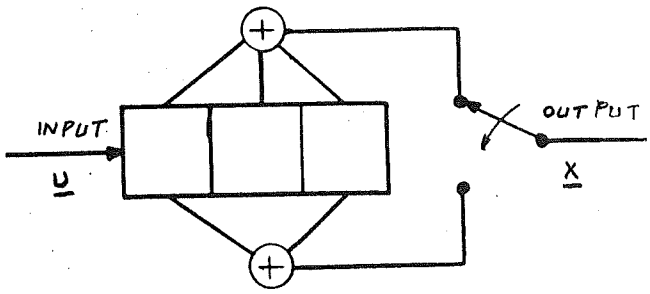

Figure 2: Codeur convolutionnel, K=3, R=$\frac{1}{2}$

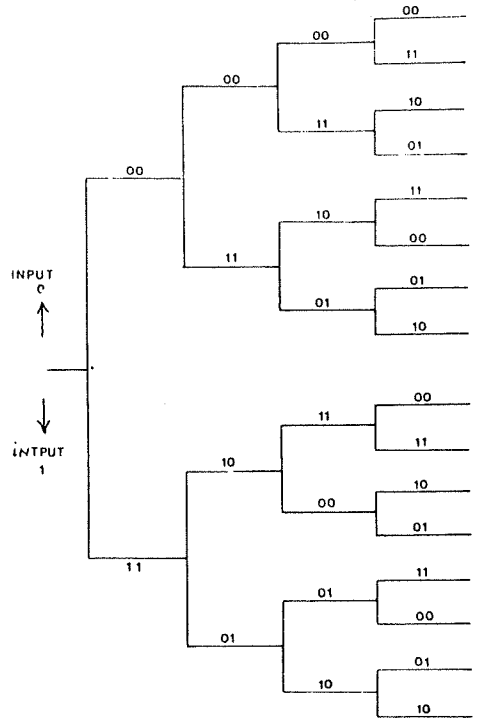

Figure 3: Arbre d'encodage du codeur de la Figure 2

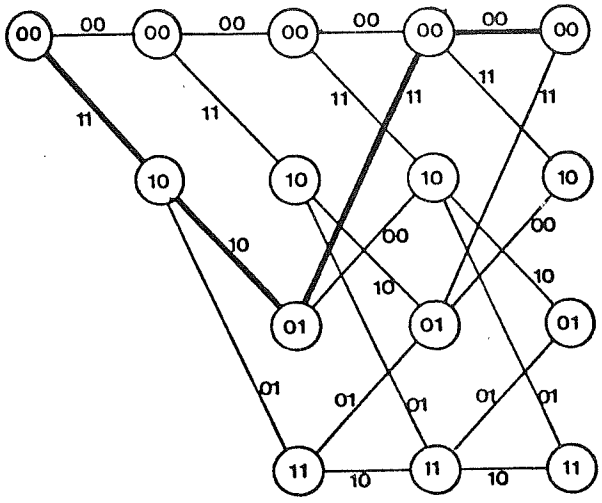

Figure 4: Treillis correspondant à l'arbre de la Figure 3

3. DÉCODAGE SÉQUENTIEL

Un décodeur séquentiel utilise la structure en arbre du code et n'explore un chemin à la fois, que la partie de l'arbre qui paraît être la plus vraisemblable sans explorer l'arbre entier. C'est donc une procédure sous-optimale. Dans un canal sans mémoire, la fonction de vraisemblance utilisée, appelée aussi "métrique de symbole" est donnée par

$$\gamma_i = \log_2 \frac{P(y_i|x_i)}{P(y_i)} - R \qquad (1)$$

où x_i est le symbole codé transmis dans le canal, y_i est le symbole reçu correspondant, R est le taux de codage et $P(y_i|x_i)$ est la probabilité de transition du canal pour des symboles x_i et y_i. Par exemple pour un code de taux $R = \frac{1}{2}$, un canal binaire symétrique de probabilité de transition p et des entrées équiprobables, la métrique (1) devient

$$\gamma_i = \begin{cases} \log_2 (2p) - \frac{1}{2} & , \quad y_i \neq x_i \\ \log_2 2(1-p) - \frac{1}{2} & , \quad y_i = x_i \end{cases} \qquad (2)$$

Dans un canal sans mémoire la métrique est additive le long des symboles des branches d'un même chemin, de sorte que la métrique $\Gamma^{(\underline{U})}$ du noeud extrêmité d'un chemin $\underline{U}$ de longueur m symboles est donnée par

$$\Gamma^{(\underline{U})} = \sum_{i=1}^{m} \gamma_i \qquad (3)$$

La métrique totale Γ tend à croître en moyenne le long du chemin correct, et tend à décroître, en moyenne le long de tous les chemins incorrects. Un exemple de cette métrique est donné à la Figure 5.

Sachant la séquence reçue du canal l'objectif du décodeur séquentiel est d'explorer l'arbre d'encodage le long du chemin ayant la métrique la plus élevée parmi tous les chemins explorés. Cependant le bruit du canal provoquant occasionnellement des chutes locales de la métrique du chemin correct, le décodeur cesse alors de suivre le chemin correct pour explorer des chemins incorrects plus vraisemblables. Par conséquent bien qu'en moyenne très faible, l'effort de décodage exprimé en nombre de calculs effectués par bit décodé est aussi très variable avec une fonction de répartition de type Pareto, c'est-à-dire:

$$P\ (c \geqslant N) \approx \ell\, N^{-\alpha}, \qquad N \gg 1 \qquad (4)$$

où ℓ est une constante, et où le paramètre α, $\alpha \geqslant 0$, appelé exposant pareto ne dépend que du taux de codage et du canal [3]-[6].

La variabilité de l'effort de calcul est l'inconvénient principal du décodage séquentiel et nécessite l'utilisation d'un tampon à l'entrée du décodeur pour y stocker les branches reçues du canal en attente d'être décodées. Le débordement de ce tampon constitue un événement d'erreur catastrophique entraînant un grand nombre de bits en erreur. Il est donc très important de réduire cette variabilité de l'effort de décodage et un certain nombre de procédures ont été élaborées à cette fin [7]-[8].

Algorithme à pile

Les deux principaux algorithmes de décodage séquentiel sont l'algorithme de Fano [4] et l'algorithme de Zigangirov-Jelinek (Z-J) [1], [3]. Le présent article ne traite que l'algorithme Z-J. Cet algorithme utilise une pile pour stocker toutes les caractéristiques des chemins explorés, et un tampon d'entrée pour stocker les séquences reçues en attente d'être décodées. Un schéma de principe du décodeur est montré à la Figure 6. La pile est une liste ordonnée où sont stockés les chemins explorés par ordre décroissant de leur métrique. Le sommet de la pile contient le chemin ayant

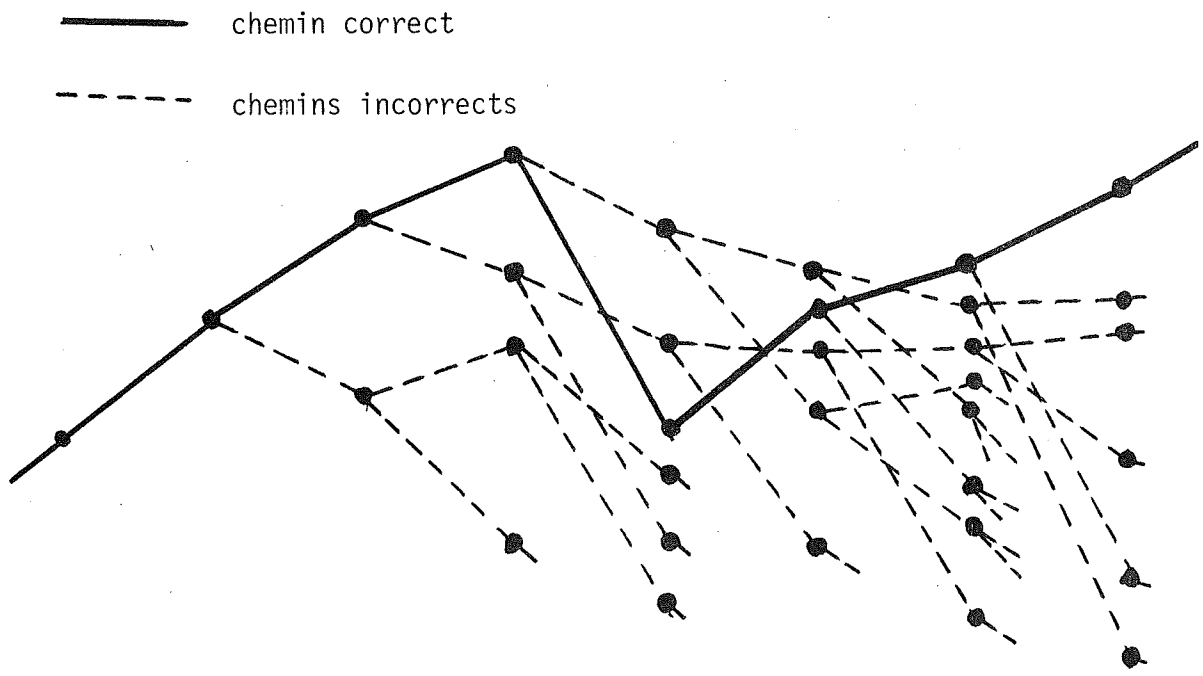

Figure 5: Exemple de métrique des chemins correct et incorrects

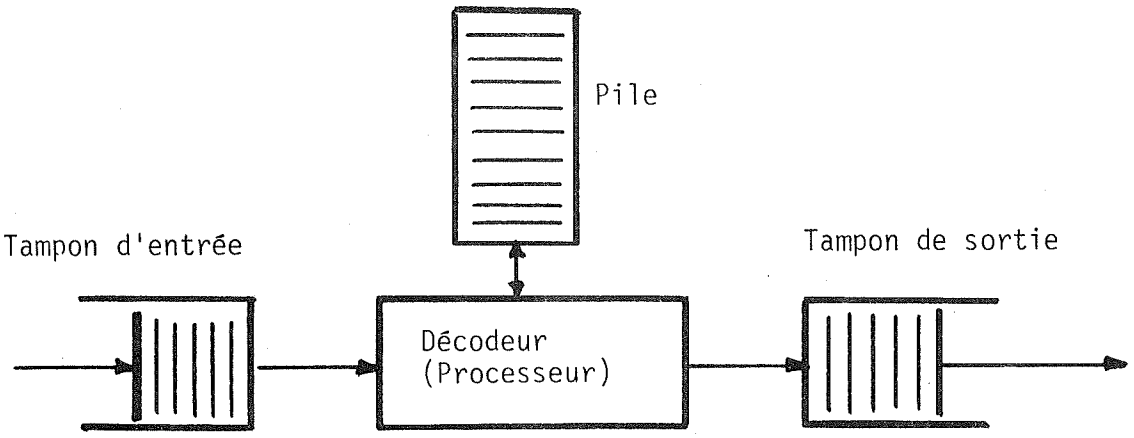

Figure 6: Schema de principe d'un décodeur à pile

la métrique maximum courante; ce chemin est donc celui qui sera prolongé. L'algorithme a donc pour objet de déterminer à chaque étape le sommet de la pile et d'en faire le prolongement. Il se compose des 3 étapes suivantes:

1. Calcul des métriques des deux chemins issus du sommet de la pile et insertion dans la pile de ces deux chemins.

2. Élimination du sommet qui vient d'être prolongé.

3. Détermination du nouveau sommet. Si c'est le noeud terminal, stop. Sinon retour à 1.

Lorsque l'algorithme arrête, le sommet de la pile est le noeud terminal du chemin décodé, qui est alors facilement récupéré.

Bien que très simple, cet algorithme n'est pas pratique car le temps nécessaire à la mise en ordre exacte de la pile est beaucoup trop élevé. Cette difficulté est contournée en effectuant une mise en ordre approximative: les noeuds explorés sont insérés aléatoirement dans des sous-piles de la façon suivante: un noeud $\underline{U}$ de métrique $\Gamma^{(\underline{U})}$ est inséré au hasard dans la sous-pile Q si

$$QH \leqslant \Gamma^{(\underline{U})} < (Q + 1)H \qquad (5)$$

où H est une valeur arbitraire.

Avec cette modification, la recherche du sommet de la pile est réduite à la recherche de la sous-pile maximum non vide, et le chemin prolongé est choisi au hasard dans cette sous-pile, habituellement selon la procédure dernier-entré-premier-sorti (LIFO). Cette modification de

l'algorithme facilite considérablement la procédure de recherche du noeud à prolonger et rend l'algorithme à pile applicable et pratique. La structure de données utilisée pour simuler l'algorithme sur ordinateur est fournie en annexe. Cette structure a servi de base à une réalisation matérielle d'un décodeur séquentiel à pile fonctionnant à environ 1 Mbit/s [9].

Un décodeur séquentiel pratique tient compte du délai de décodage variable qui découle de la variabilité de l'effort de décodage en utilisant un tampon d'entrée et un tampon de sortie (voir Figure 6). Le tampon de sortie régularise le débit de sortie des séquences décodées alors que le tampon d'entrée sert à stocker les données provenant du canal et qui attendent d'être décodées. Aussi un problème important concerne le débordement de ces tampons. On peut montrer que quelle que soit sa taille, il existe une probabilité non nulle pour que le tampon d'entrée déborde entraînant une perte de données et une rupture du lien de communication. L'analyse des débordements et des procédures de redémarrage du système sont parmi les problèmes importants de décodage séquentiel [10]-[11].

Afin de réduire les conséquences d'un débordement et d'une rupture du lien de communication, les données sont généralement organisées en "blocs" comportant quelque 500 à 2000 branches, chaque bloc se terminant par une séquence connue appelée "queue du message". La queue de longueur égale à (K-1) branches permet de remettre à zéro le registre à décalage du codeur local du décodeur, et de resynchroner le système. En cas de débordement, le bloc en question est éliminé, le système est remis à zéro et le décodage peut se poursuivre pour les blocs suivants.

Effort de calcul du décodeur séquentiel

Quel qu'en soit l'algorithme, le décodage séquentiel implique toujours la possibilité pour le décodeur de revenir en arrière dans l'arbre et de changer une décision antérieure, c.a.d. k de prendre une autre alternative que celle qui semblait être la meilleure. D'un point de vue théorique et analytique, chaque opération de prolongation d'un chemin est définie comme étant un "calcul". Comme le nombre d'extensions effectuées est aléatoire, le nombre de calculs effectués par bloc décodé est donc également aléatoire. Aussi, contrairement aux algorithmes de décodage déterministe, l'analyse du décodage séquentiel concerne aussi bien la performance d'erreurs que la distribution de l'effort de calcul. Cette variabilité de l'effort de calcul est l'un des principaux inconvénients du décodage séquentiel et le problème de sa diminution a été l'objet d'une grande activité de recherche [7]-[8].

Une analyse théorique relativement complexe a montré que le nombre de calculs C effectué par bit décodé a une fonction de distribution cumulative qui suit asymptotiquement une loi Pareto, donnée par (4) et répétée ci-dessous:

$$P\ (C \geqslant N) \leqslant \ell N^{-\alpha}\ , \qquad N \gg 1 \tag{6}$$

où ℓ et α dépendent du canal et du taux de codage R. L'exposant α est appelé l'exposant Pareto et est un des paramètres clef pour évaluer la performance et la conception de décodeur séquentiel.

La cumulative (6) indique que l'effort de calcul suit une décroissance algébrique et non pas exponentielle avec N, représentant ainsi un autre inconvénient du décodeur séquentiel. Par conséquent, il devient impératif de s'assurer que le nombre moyen de calculs par bit soit fini, et également de faire face, en pratique, au retard qui découle de la variabilité de l'effort du décodage.

Tel que mentionné plus haut, le retard de décodage se règle par l'utilisation de tampons d'entrée et de sortie. Quant à borner le nombre moyen de calculs, la réponse réside dans l'analyse théorique de la variabilité de l'effort de calcul. Une analyse des moments de la distribution [6] a montré que si l'exposant Pareto α est inférieur à 2, la variance de l'effort de décodage diverge, et si $\alpha \leq 1$, la moyenne de cet effort n'est plus bornée, c.a.d. le nombre moyen de calculs devient théoriquement infini. Ceci se traduit en pratique par un décodage erratique, avec de très longues recherches arrière dans l'arbre et des débordements des tampons et de la pile. Le taux de codage qui correspond à la valeur limite $\alpha = 1$ est appelé taux de coupure ("Computational Cut Off Rate") et est dénoté R_{comp}.

Ce taux R_{comp} ne dépend que du canal et se calcule facilement [3], mais d'un point de vue pratique, ce paramètre représente la limite extrême d'utilisation de décodeurs séquentiels. Aussi un important paramètre de design est le rapport R/R_{comp}, que l'on désire aussi près que possible de 1 mais sans l'atteindre. En pratique on choisi des points d'opération où R/R_{comp} est compris entre 0.80 et 0.99, et on évalue approximativement l'exposant Pareto par

$$\alpha \approx \frac{R_{comp}}{R} \qquad (7)$$

On peut noter en passant que la valeur de E_b/N_o correspondant à R_{comp} peut être calculée pour différents modèles de canaux et de niveaux de quantification, avec en général une amélioration de 2 dB lorsque la quantification du canal passe de 2 niveaux (1 bit) à 8 niveaux (3 bits) [3]-[12].

L'analyse théorique des moments de l'effort de calcul en général, et du nombre moyen de calculs $\bar{C}$ en particulier est réputée être difficile, et ne donne que des bornes asymptotiques et relativement peu serrées sur des ensembles de codes. D'un point de vue pratique, ces bornes sont donc très peu utiles pour des fins de design impliquant un code particulier. On doit

donc avoir recours à de longues simulations sur ordinateur pour obtenir des résultats utilisables. Cependant, utilisant une approche théorique totalement différente et basée sur les processus de ramification, une analyse récente de $\bar{C}$ a donné des résultats considérablement plus précis [13]. De plus, les résultats de cette analyse sont directement applicables au cas particulier car ils utilisent les paramètres du code ainsi que les valeurs particulières des métriques utilisées par le décodeur.

La nature Pareto de la distribution de l'effort de calcul a été confirmée pour toutes sortes de canaux. Un raisonnement simple permet d'expliquer un tel comportement. Lorsque le bruit dans le canal devient assez fort pour provoquer une chute de la métrique du chemin correct, le décodeur entre dans une phase de recherche arrière et prolonge les noeuds des chemins incorrects en conformité avec l'algorithme. Le nombre de ces chemins incorrects croît de façon exponentielle avec la profondeur de la chute de métrique du chemin correct. Cependant, pour des canaux sans mémoire, tout intervalle de bruit qui provoque une chute de métrique apparaît avec une probabilité qui décroît exponentiellement avec la durée de cet intervalle. Le comportement Pareto n'est rien d'autre que l'effet combiné de ces deux comportements exponentiels.

Problèmes de débordement

Il est clair qu'un débordement du tampon d'entrée aurait des conséquences catastrophiques. Il est donc impératif de prévoir une taille de tampon adéquate pour minimiser, voire éliminer un tel événement.

Cependant on montre que quelle que soit sa taille, il existe toujours une probabilité non nulle pour que le tampon d'entrée déborde. Cette probabilité est assez grande (nettement plus grande que la probabilité d'erreur) et un débordement conduit à des effacement (en anglais "erasures") Ces effacements ne sont pas des erreurs mais sont considérés plutôt comme étant des incertitudes sur la valeur des bits décodés.

Pour des séquences de longueurs L bits, la probabilité de débordement du tampon d'entrée est approximée par

$$P \text{ (débordement)} \approx L(GB)^{-\alpha} \qquad (8)$$

où B est la taille du tampon, α est l'exposant Pareto et G est le gain de vitesse du décodeur [3]. Ce gain de vitesse est le rapport entre le temps d'interarrivée des bits du canal et le temps d'un calcul par le décodeur. L'expression (8) indique encore une distribution Pareto et une probabilité de débordement qui ne varie que lentement avec la taille du tampon, donc très difficile à combattre. Aussi en pratique, on choisit des tampons d'entrée très grands, et on sectionne les séquences d'information en blocs de longueurs variant de 500 à 2000 ou 3000 bits. De plus, on prend toujours certaines procédures de recouvrement en cas de débordement. Ces procédures peuvent consister en une demande de retransmission des blocs qui sont sur le point de déborder [11], [14], [15], ou en un arrêt de décodage proprement dit et en une estimation de la séquence transmise. Ces procédures de recouvrement, qui peuvent varier selon les applications, sont une partie essentielle du décodage séquentiel et ne peuvent être ignorées dans une réalisation pratique.

4. VARIANTES DE L'ALGORITHME Z-J

Des recherches antérieures [7]-[16] ont montré que la variabilité de l'effort de calcul d'un décodeur séquentiel peut être réduite en faisant non pas l'extension d'un seul chemin (le sommet de la pile), mais l'extension simultanée d'un certain nombre M de chemins les plus vraisemblables, c.a.d. ayant les métriques les plus élevées. Ce nombre M peut ne pas être fixe, et peut même s'adapter aux besoins courants imposés par le bruit dans le canal de transmission. Ces variantes du décodage séquentiel à extension multiples appartiennent à la classe d'algorithmes généralisés à pile qui unifient les techniques de décodage séquentiel et de Viterbi [7]. Dans toutes ces variantes, la réduction de la variabilité de calcul est toujours obtenue au prix d'un effort moyen plus grand, mais aussi sans dégradation de la performance d'erreur. De plus, étant donné la structure de la pile, leur mise en oeuvre est particulièrement facile, n'impliquant par rapport à l'algorithme de base qu'une complexité additionnelle minime. Les cas particuliers des variantes multichemins de l'algorithme Z-J considérées ici sont brièvement décrits ci-dessous.

i) Algorithme M chemins

Il s'agit tout simplement de faire l'extension simultanée des M chemins occupant les plus hautes positions dans la pile. Cet algorithme peut être très facilement rendu adaptatif en imposant des restrictions sur M, ou en modifiant M en fonction du comportement général de la métrique du sommet de la pile [7].

ii) Algorithme S sous-piles

Cet algorithme fait l'extension simultanée de tous les chemins contenus dans les S sous-piles de métrique maximum courante [16]. Cet algorithme est auto adaptatif car le nombre de chemins résidant dans les sous-piles varie en fonction du bruit dans le canal. Lorsque le canal est calme, les sous piles maximum ne contiennent que quelques chemins et sont

souvent vides, alors que pendant des périodes de bruit intense les métriques ayant une tendance à avoir des valeurs très rapprochées les unes des autres, et les populations de chemins dans les mêmes sous-piles augmentent considérablement. Là encore S peut ne pas être maintenu constant tout au long du décodage, et pratiquement on impose toujours un nombre maximum de chemins à prolonger simultanément. La Figure 7 donne les distributions de l'effort de calcul des algorithmes Z-J, de l'algorithme 4- chemins et de l'algorithme S sous-piles dans lequel S est égal à 5, 10 et 12. On peut voir que l'algorithme 4- chemins améliore de façon substantielle la distribution de l'algorithme Z-J de base, et que la variabilité de l'effort de calcul est encore réduite par l'algorithme S sous-piles. À mesure que S augmente la cumulative se rapproche d'une fonction échelon, et en particulier, la Figure 7 montre que pour S = 12 le nombre maximum de calculs pour décoder un bit se chiffre à 34, et que cet événement est très rare, n'étant apparu qu'avec une probabilité de 1×10^{-5}. Quant à l'effort de calculs moyen il augmente proportionnellement avec S, ne valant que 3.45 pour S = 12, alors qu'il est égal à 1.13 et 4.05 pour les algorithmes Z-J et 4- chemins respectivement. Enfin on remarque que l'augmentation du nombre d'extensions simultanées améliore aussi la performance d'erreurs, ce qui bien sûr était prévisible.

iii) Algorithme adaptatif

Cet algorithme tend à adapter le nombre de chemins prolongés simultanément avec le comportement de la métrique maximum, c.a.d. du courant dans le canal. Pour cela on observe la croissance de cette métrique maximum. Si elle croît, un seul chemin est prolongé. Si elle chute d'une valeur D, on prolonge simultanément un nombre M(D), où M(D) est une fonction non décroissante de D [7]. La Figure 8 donne la distribution du nombre de calculs par bit pour quelques fonctions linéaires de M(D). Là encore on peut voir l'effet de l'augmentation du nombre d'extensions simultanées sur la fonction de répartition de l'effort de calcul et sur la probabilité d'erreur.

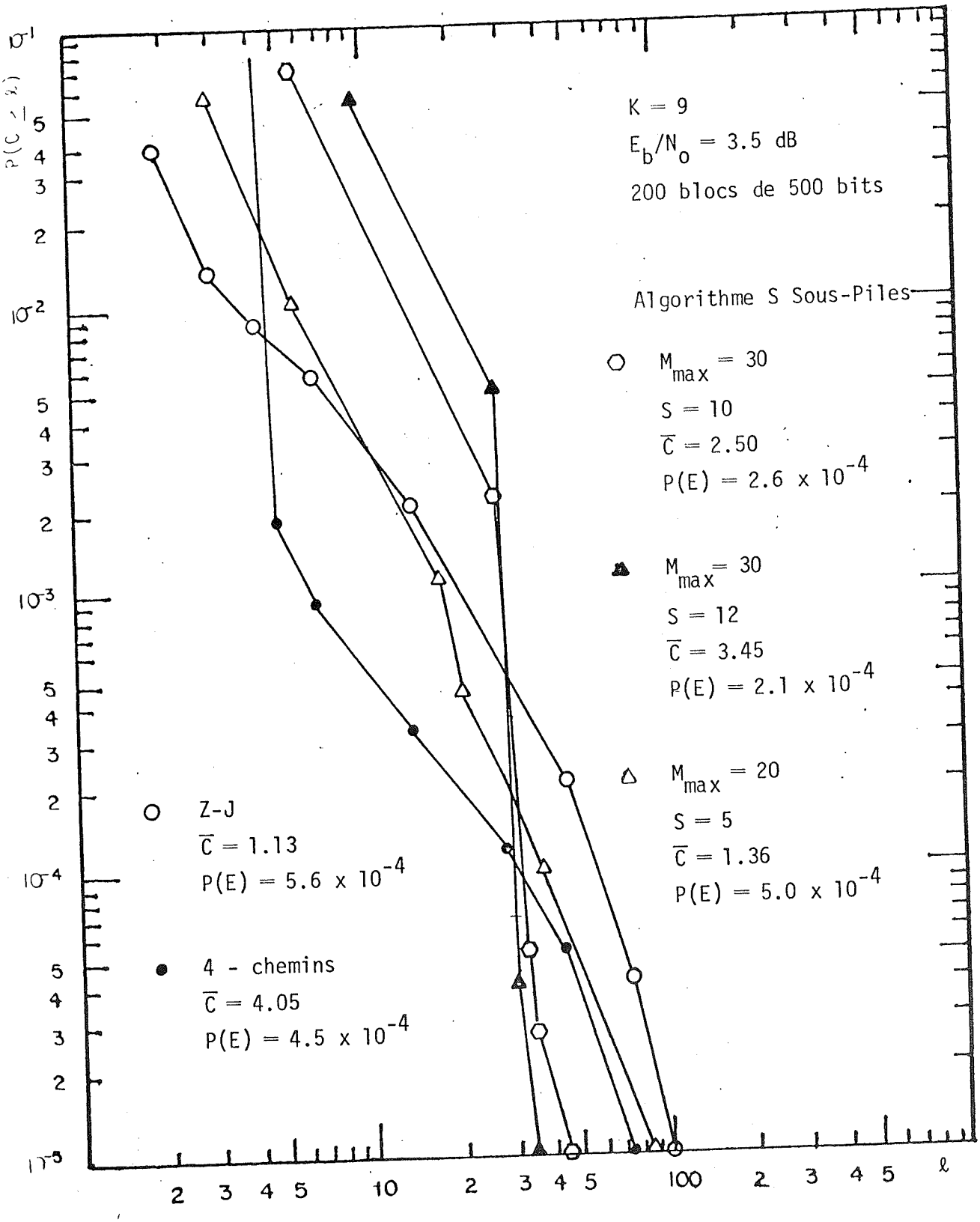

Figure 7: Cumulative du nombre de calculs par bit, Algorithme s-sous piles

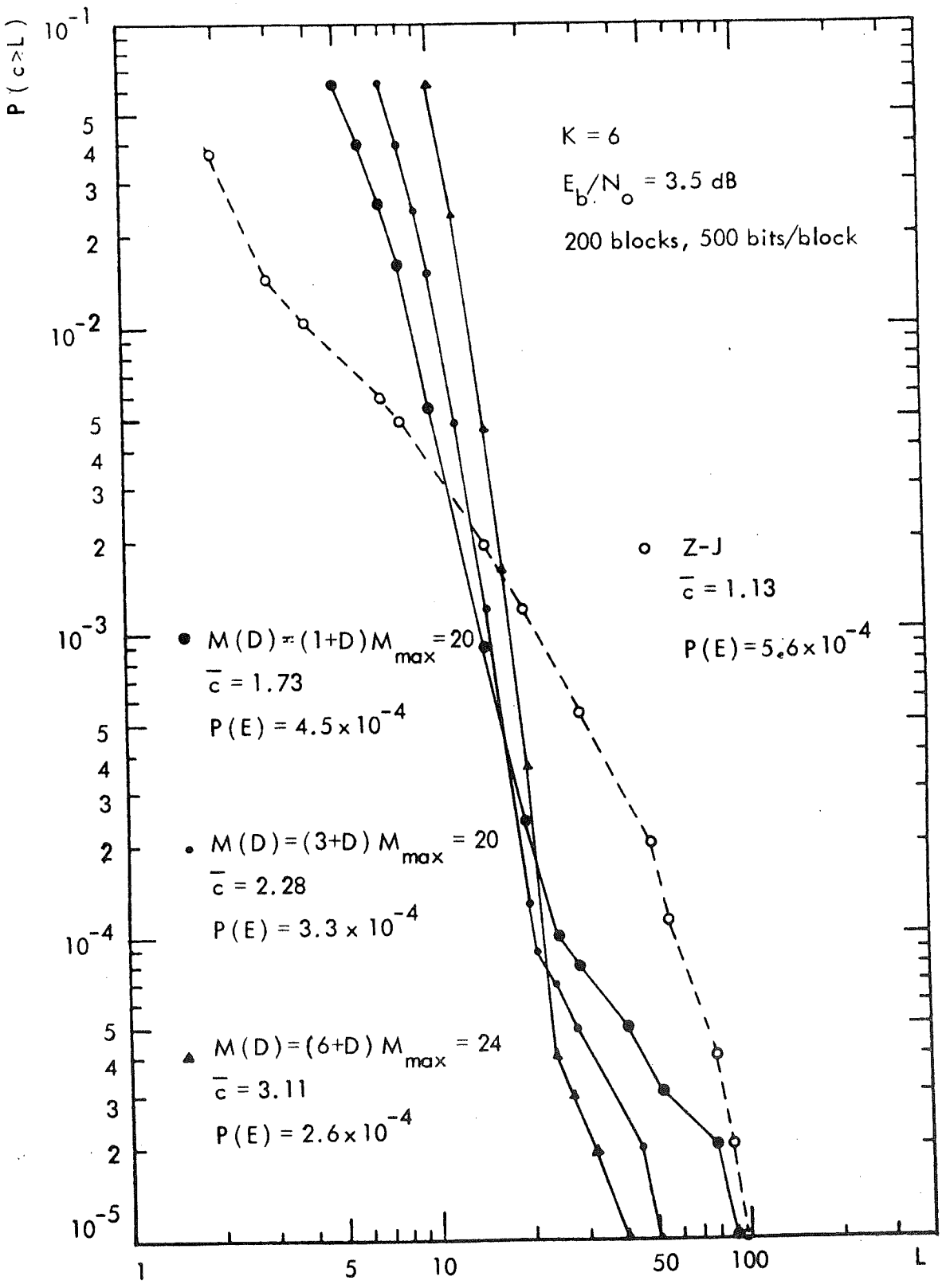

Figure 8: Cumulative du nombre de calculs par bit, Algorithme Adaptatif

iv) Algorithme prédicteur

Il s'agit ici de déterminer à l'avance la chute D de la métrique afin de mieux faire varier N ou M(D) des algorithmes ii) ou iii). Pour cela la chute de la métrique maximum est déterminée en pénétrant l'arbre le long du chemin de plus vraisemblable sur une fenêtre d'observation de P branches. Utilisant les variations de la métrique dans cette fenêtre, l'algorithme prolonge alors un nombre de chemins qui tient compte non seulement du comportement passé de la métrique mais aussi de son comportement futur. La Figure (9) illustre cet algorithme utilisé en conjonction avec l'algorithme S sous-piles. La distribution de L'effort de calculs est encore améliorée par rapport à l'algorithme Z-J, mais par rapport à l'algorithme S sous-piles seul, le principal avantage de l'algorithme prédicteur réside en la diminution du nombre moyen de calcul. Cependant, la mise en oeuvre de la fonction de prédiction de l'algorithme est quelque peu plus complexe.

Comme on l'a vu, dans toutes ces variantes la réduction de l'effort de calcul est obtenue au prix d'un effort moyen plus élevé. aussi faut-il encore évaluer l'impact de ces algorithmes sur les tailles et sur les comportements de la pile et de la file d'attente à l'entrée du décodeur. Ces problèmes sont examinés ci-après.

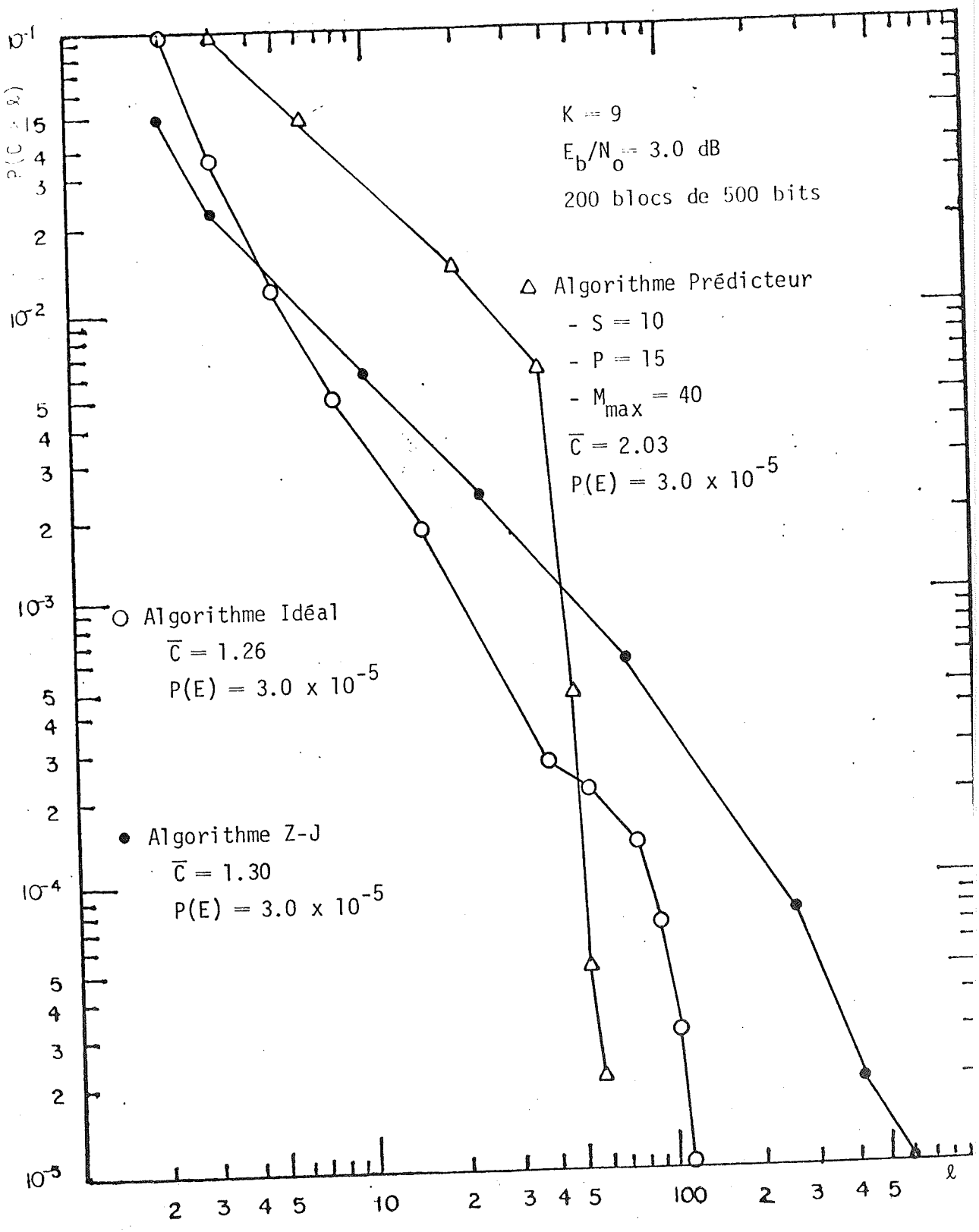

Figure 9: Cumulative du nombre de calculs par bit, Algorithme Predicteur

5. IMPACT SUR LES TAILLES DE LA PILE ET DU TAMPON D'ENTRÉE

L'algorithme Z-J et ses variantes décrites ci-dessus ont été simulés sur ordinateur, et les informations pertinentes sur le remplissage de la pile et sur la file d'attente à l'entrée du décodeur ont été recueillies. Les simulations ont été effectuées en utilisant un code de Johannesson [17] de taux $R = \frac{1}{2}$ et de longueur de contrainte $K = 24$. Dans ces simulations les données sont transmises sous forme de blocs de longueur $L = 500$ bits auxquels on ajoute une queue de 23 bits pour fins de synchronisation. Les symboles transmis sont reçus en présence de bruit blanc gaussien, en quantification ferme avec rapports signaux à bruit E_b/N_o égaux à 4.64 dB et 5.60 dB, correspondant à des rapports R/R_{comp} égaux à 0.99 et 0.85 respectivement.

Remplissage de la pile

Le nombre de noeuds stockés dans la pile, le nombre de débordements de la pile et le nombre d'erreurs sont observés à la fin du décodage de chaque bloc. La fonction de répartition du nombre d'entrées dans la pile pour l'algorithme Z-J est montrée à la Figure 10, et indique une distribution de type Pareto. Ce même type de distribution a été observé pour tous les algorithmes étudiés et est la conséquence directe de la distribution Pareto de l'effort de calcul du décodeur séquentiel. En effet comme chaque extension, simple ou multiple implique au moins une entrée dans la pile, il est clair que la distribution du nombre d'entrées devra suivre celle du nombre de calculs tout en tenant compte du nombre d'extensions simultanées effectuées par l'algorithme particulier. Ainsi, il a été observé que pour une même valeur de E_b/N_o, plus le nombre d'extensions simultanées, et donc le nombre moyen de calculs $\overline{C}$, est élevé, plus l'exposant, Pareto α (c.a.d. la pente de la courbe) est élevé. Les résultats des simulations concernant la pile sont résumés aux Tableaux 1 et 2 pour R/R_{comp} valant 0.85 et 0.99 respectivement. On y voit en particulier la corrélation très nette entre l'augmentation du nombre de calculs moyen $\overline{C}$, l'exposant Pareto de la pile et

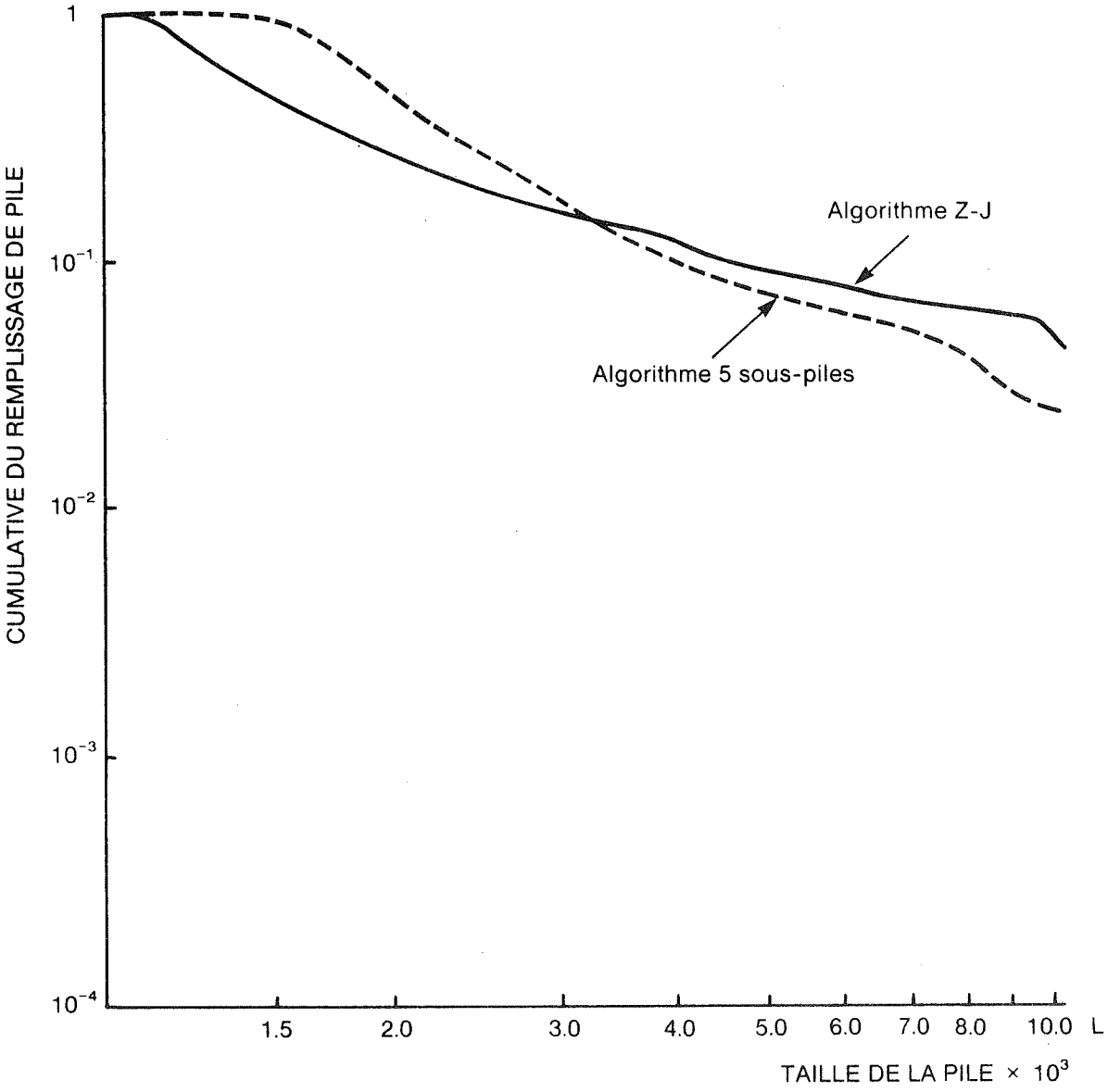

Figure 10: Fonction de répartition du remplissage de la pile pour les algorithmes Z-J et 5 sous-piles.

la taille moyenne de la pile. Les Tableaux 1 et 2 donnent aussi pour chaque algorithme la taille minimum de la pile pour une probabilité de débordement égale à 1×10^{-1}, 1×10^{-2} et 1×10^{-3}. La Figure 11 reprend ces valeurs et donne pour R/R_{comp} =0.99 la probabilité de débordement des différents algorithmes en fonction de la taille de la pile. On remarque que pour des probabilités de débordement élevées, comprises entre 1×10^{-1} et 1×10^{-2}, l'algorithme Z-J est sensiblement équivalent aux variantes multichemins. Cependant pour de plus faibles probabilités de débordement, l'algorithme Z-J requiert considérablement plus de mémoire pour sa pile que les variantes multi chemins. Naturellement cette situation n'est que la conséquence d'un exposant Pareto plus faible pour l'algorithme Z-J.

À la lumière de ces résultats on peut tirer les conclusions suivantes: pour des probabilités de débordement de pile de l'ordre de 10^{-2} à 10^{-3}, et un canal faiblement bruité une valeur R/R_{comp} faible, de l'ordre de 0.85, les algorithmes multi chemins ne sont pas très avantageux, et un choix judicieux serait l'algorithme Z-J. Cependant pour des canaux fortement bruités avec R/R_{comp} très prêt de 1, les algorithmes multichemins sont préférables à l'algorithme Z-J. On choisira probablement l'algorithme fournissant l'exposant Pareto le plus élevé sous les contraintes pratiques imposées par l'application particulière.

Si d'autre part on ne disposait que d'une pile de petite taille (environ 5000 entrées), alors il est préférable de choisir un algorithme dont le nombre de calculs est relativement faible (par exemple l'algorithme Z-J) bien que ce choix puisse entraîner une variabilité de calcul appréciable. Par contre si la pile est de grande taille par exemple supérieure à 10000 ou 20000 noeuds, les algorithmes multichemins sont alors plus indiqués que l'algorithme Z-J.

TABLEAU 1: REMPLISSAGE DE LA PILE

R/R_{comp} = 0.85, Eb/No = 5.60 dB, K = 24, R = ½

500 blocs simulés, 523 bits/bloc

ALGORITHME / PARAMÈTRE	Z-J	2-CHEMINS	N SOUS-PILES		ADAPTATIF
			N = 5	N = 7	M(D) = 1+½ D
ERREURS	0	0	0	0	0
DÉBORDEMENTS PILE, S = 10^4	0	0	0	0	0
EXP. PARETO α (PILE)	2.8	3.5	3.5	4.2	3.5
CALCULS MOYENS $\overline{C}$	1.10	1.82	1.24	1.58	1.40
PILE MOYENNE $\overline{S}$	1148	1905	1288	1660	1480
*PILE MINIMUM $S_{min}^{(1)}$	1260	2000	1480	1950	1620
*PILE MINIMUM $S_{min}^{(2)}$	2500	3160	2570	2950	2750
*PILE MINIMUM $S_{min}^{(3)}$	5600	6000	4900	5250	5250

Note * PILE MINIMUM $S_{min}^{(k)} = S_{min}$: $Pr\ (S \geq S_{min}) = 10^{-k}$, k = 1,2,3

TABLEAU 2: REMPLISSAGE DE LA PILE

R/R_{comp} = 0.99, Eb/No = 4.64 dB, K = 24, R = ½

200 blocs simulés, 523 bits/bloc

ALGORITHME / PARAMÈTRE	Z-J	2-CHEMINS	N SOUS-PILES		ADAPTATIF
			N = 5	N = 7	M(D) = 1+½ D
ERREURS	0	0	0	0	0
DÉBORDEMENTS PILE, S = 10^4	2	2	2	2	2
EXP. PARETO α (PILE)	1.37	1.80	1.80	2.63	2.10
CALCULS MOYENS $\overline{C}$	1.67	2.45	2.19	2.88	2.20
PILE MOYENNE $\overline{S}$	1700	2570	2290	2950	2290
*PILE MINIMUM $S_{min}^{(1)}$	2340	3020	2950	3980	2880
*PILE MINIMUM $S_{min}^{(2)}$	10,000	10,000	8900	10,000	8900
*PILE MINIMUM $S_{min}^{(3)}$	53700	35500	31600	24000	27000

Note * PILE MINIMUM $S_{min}^{(k)} = S_{min}$: $Pr\ (S \geq S_{min}) = 10^{-k}$, k = 1,2,3

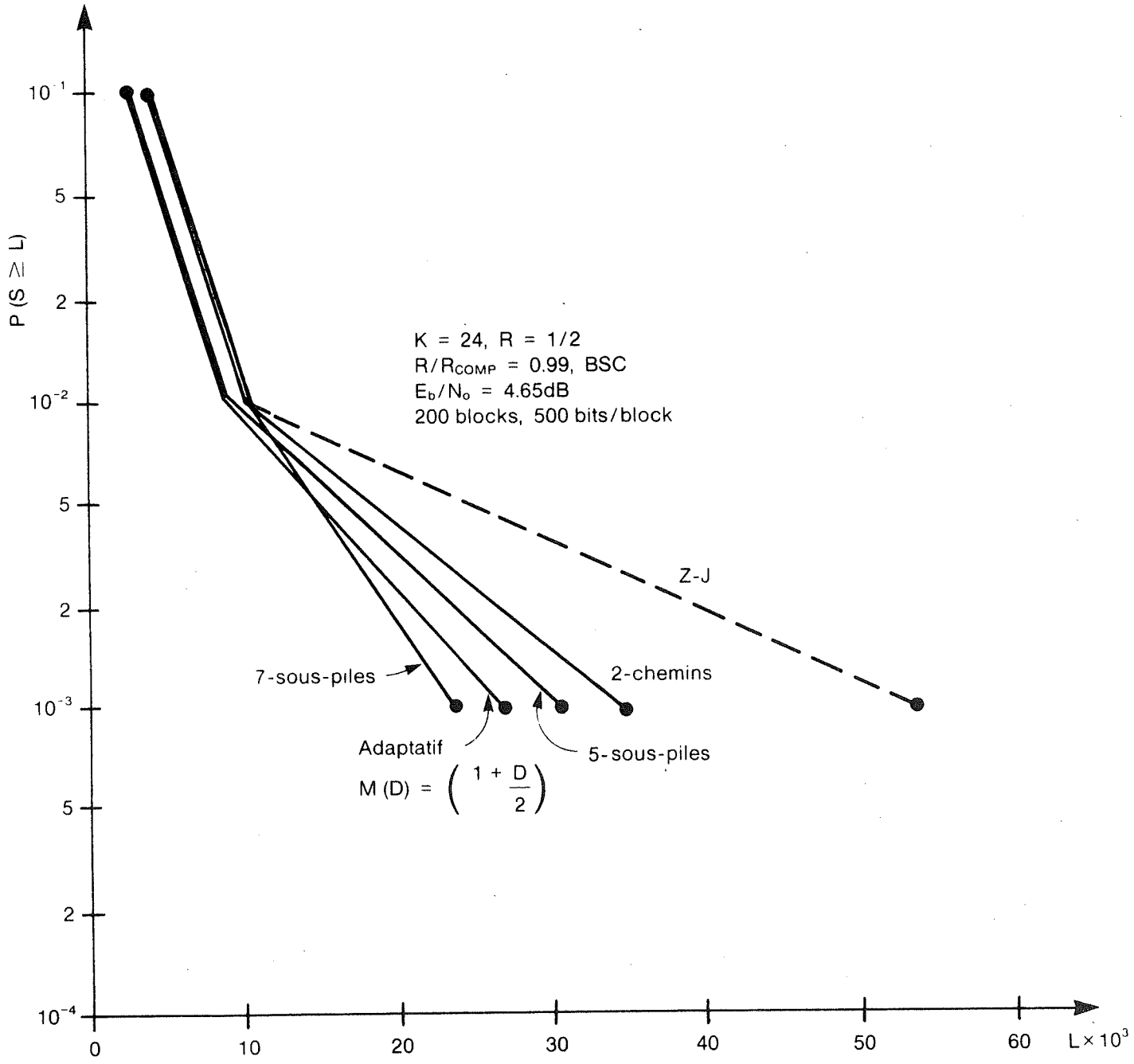

Figure 11: Cumulative du remplissage de la pile pour plusieurs algorithmes.

File d'attente dans le tampon d'entrée

Quel que soit l'algorithme de décodage séquentiel la variabilité de l'effort de calcul nécessite l'utilisation d'un tampon à l'entrée du décodeur afin d'y stocker les branches reçues du canal pendant les recherches arrière du décodeur. Un comportement typique de la file d'attente dans ce tampon est montré à la Figure 12. Lorsque le décodeur entre dans une phase de recherche arrière d'un meilleur chemin, les symboles codés reçus du canal s'accumulent dans le tampon d'entrée et forment une file d'attente. Dès que le décodeur trouve le bon chemin sa progression en avant doit être rapide afin de rattraper en quelque sorte le "temps perdu"; un décodeur séquentiel doit donc bénéficier d'un gain de vitesse par rapport au taux de réception des branches du canal. Ce gain de vitesse G qui est le nombre de calculs que le décodeur peut effectuer durant le temps de réception d'une nouvelle branche du canal, doit nécessairement être supérieur à l'effort de calcul moyen $\overline{C}$. Si non, en moyenne la file d'attente croîtra sans cesse et conduira avec certitude à un débordement du tampon d'entrée. Par exemple sur la Figure 12, en fonction du temps la file d'attente apparaît sous la forme de pics disjoints, indiquant un gain de vitesse adéquat.

Dans les décodeurs séquentiels à pile la dynamique de la file d'attente est dépendante de la taille de la pile. En effet une pile de très grande taille permet des recherches arrières importantes sans débordements de la pile, ce qui peut entraîner un accroîssement considérable de la file d'attente dans le tampon d'entrée, voire même un débordement de ce tampon. De plus, la taille de la file d'attente dépend aussi du gain de vitesse du décodeur. Par conséquent le comportement de la file d'attente dépend de la taille de la pile et du gain de vitesse. L'analyse suivante permet de voir que sous certaines conditions un tampon d'entrée de taille maximum égale à 2 longueurs de bloc ne débordera jamais.

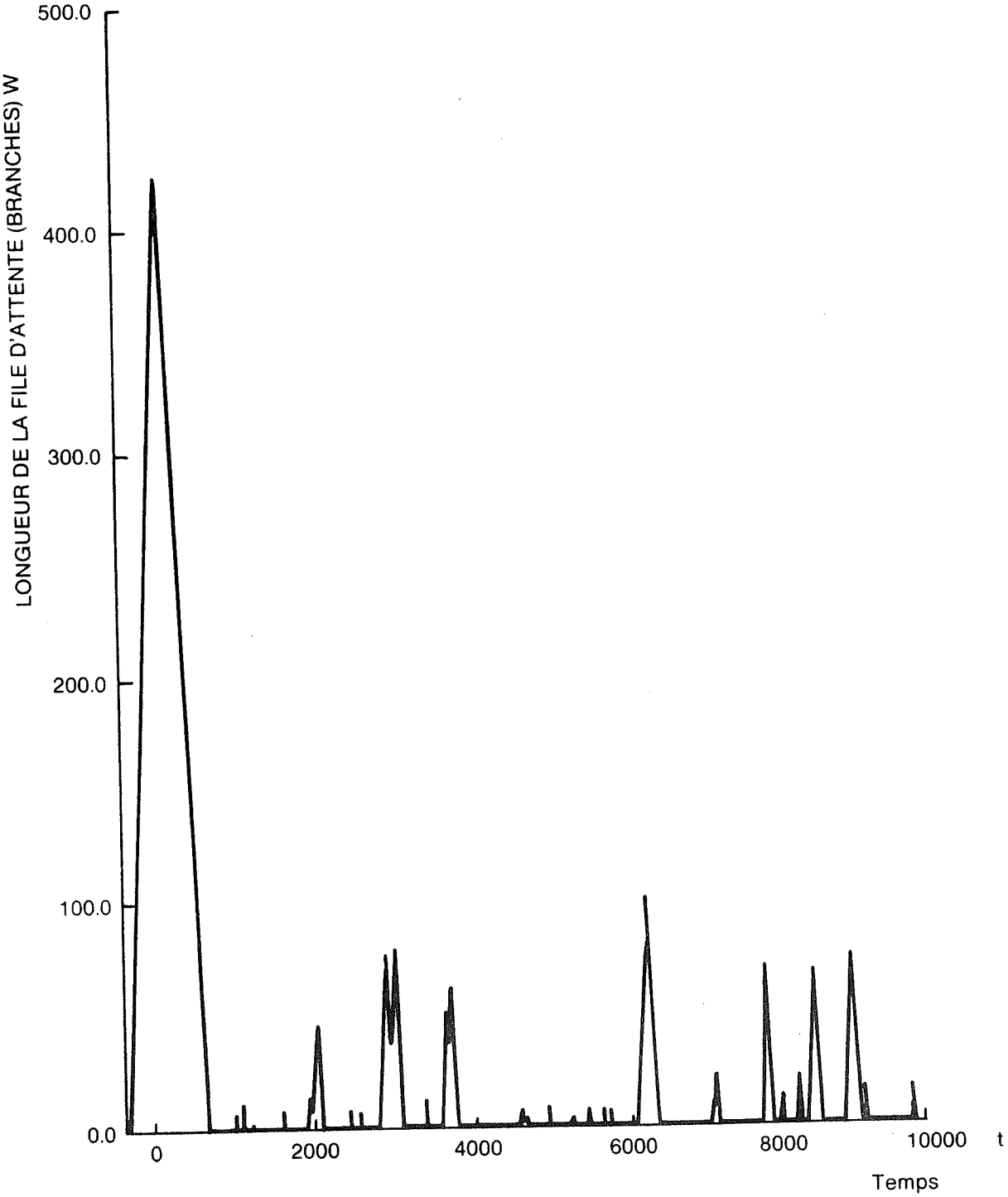

Figure 12: Dynamique de la file d'attente dans le tampon d'entrée. (Le temps est en unité d'interarrivées des branches).

Soit G, le gain de vitesse du décodeur, S la taille de la pile et L la longueur des blocs en bits. Supposons le tampon d'entrée vide à l'arrivée de la première branche, c.a.d. du premier bit d'information du bloc courant N, et supposons que les (L-1) premiers bits d'information reçus soient décodés en (L-1) calculs. Ces (L-1) bits occupent donc 2(L-1) places dans la pile. Supposons à présent que le L^e= et dernier bit du bloc soit très difficile à décoder et que son décodage nécessite toutes les [S - 2(L-1)] places restantes dans la pile. À la fin du décodage de ce dernier bit la taille en bits de la file d'attente W_N dans le tampon d'entrée est égale à

$$W_N = [S - 2(L-1)] / 2G \tag{9}$$

Pour le décodage du bloc suivant, (N+1), considérons le cas extrême où le premier bit provoque un débordement de la pile. Pendant le temps nécessaire à ce débordement un nombre (S / 2G) de branches supplémentaires viennent s'ajouter à la file d'attente dans le tampon. Au débordement du bloc N+1, la file d'attente devient alors

$$W_{N+1} = W_N + (S / 2G) = (S - L+1) / G \tag{10}$$

La pile ayant débordé pour le bloc (N+1), le décodeur élimine du tampon d'entrée tous les bits correspondant à ce bloc, c'est-à-dire au plus L bits. Après l'élimination des L bits, pour le début du bloc suivant, (N+2), la file d'attente devient

$$W_{N+2} = W_{N+1} - L = W_N + (\beta-1) L \tag{11}$$

où
$$\beta = \frac{S}{2 GL} \tag{12}$$

Ici βL représente le nombre maximum de branches appartenant à un même bloc qui s'ajoutent à la file d'attente lorsque ce bloc provoque un débordement de la pile. L'équation (11) indique que dépendant de la valeur de β, la file d'attente sera bornée ou non pour les débordements subséquents de la pile, c.a.d. pour les blocs (N+2), (N+3) etc. On considère deux cas:

i) $\underline{\beta \geq 1}$

Si le bloc (N+2) fait déborder la pile, alors utilisant le même raisonnement que pour (11) la file d'attente au début du bloc (N+3) devient

$$W_{N+3} = W_{N+2} - L + \frac{S}{2G} = W_{N+2} + (\beta-1)\, L \qquad (13)$$

utilisant (11) on obtient

$$W_{N+3} = W_N + 2\,(\beta-1)\, L \qquad (14)$$

Si les blocs suivants provoquent aussi un débordement de la pile, on obtient alors pour la file d'attente

$$W_{N+4} = W_{N+3} + (\beta-1)\, L = W_N + 3\,(\beta-1)\, L \qquad (15)$$

et en général, au début du bloc (N+m) la file d'attente est

$$W_{N+m} = W_N + (m-1)\,(\beta-1)\, L \;,\; m = 1,2,3... \qquad (16)$$

L'équation (16) indique que sous ces conditions la file d'attente n'est pas bornée et conduira à un débordement du tampon d'entrée quelque soit sa taille. Examinons à présent l'autre cas,

ii) $\beta < 1$

Dans ce cas le nombre maximum de nouvelles entrées dans la file d'attente lors d'un débordement de la pile est égal à

$$\frac{S}{2\,G} = \beta L < L \tag{17}$$

Le décodeur élimine donc au plus βL bits du tampon, et la longueur de la file d'attente lors d'un débordement de la pile est:

$$W_{N+1} = W_N + \beta L \tag{18}$$

utilisant (9) et (12), cette expression devient

$$W_{N+1} = 2\,\beta L - 2\,\frac{(L-1)}{S}\,\beta L \tag{19}$$

c'est-à-dire

$$W_{N+1} < 2\,\beta L \tag{20}$$

Comme ceci se répète pour tous les autres blocs qui peuvent faire déborder la pile, alors la longueur de la file d'attente est bornée par

$$W_{max} < 2\,\beta L < 2\,L \tag{21}$$

Par conséquent pour $\beta < 1$ ou encore $G \geq \frac{S}{2\,L}$ un tampon d'entrée de longueur maximum 2 L branches ne débordera jamais car pendant le temps nécessaire au débordement de la pile, la file d'attente du tampon d'entrée ne peut croître de plus de βL branches qui sont d'ailleurs ensuite expurgées. Sous cette condition, naturellement, la distribution de la file d'attente ne sera pas de type asymptotiquement Pareto. Tel que montré à la Figure 13, la queue de la distribution tend à décroître à la verticale.

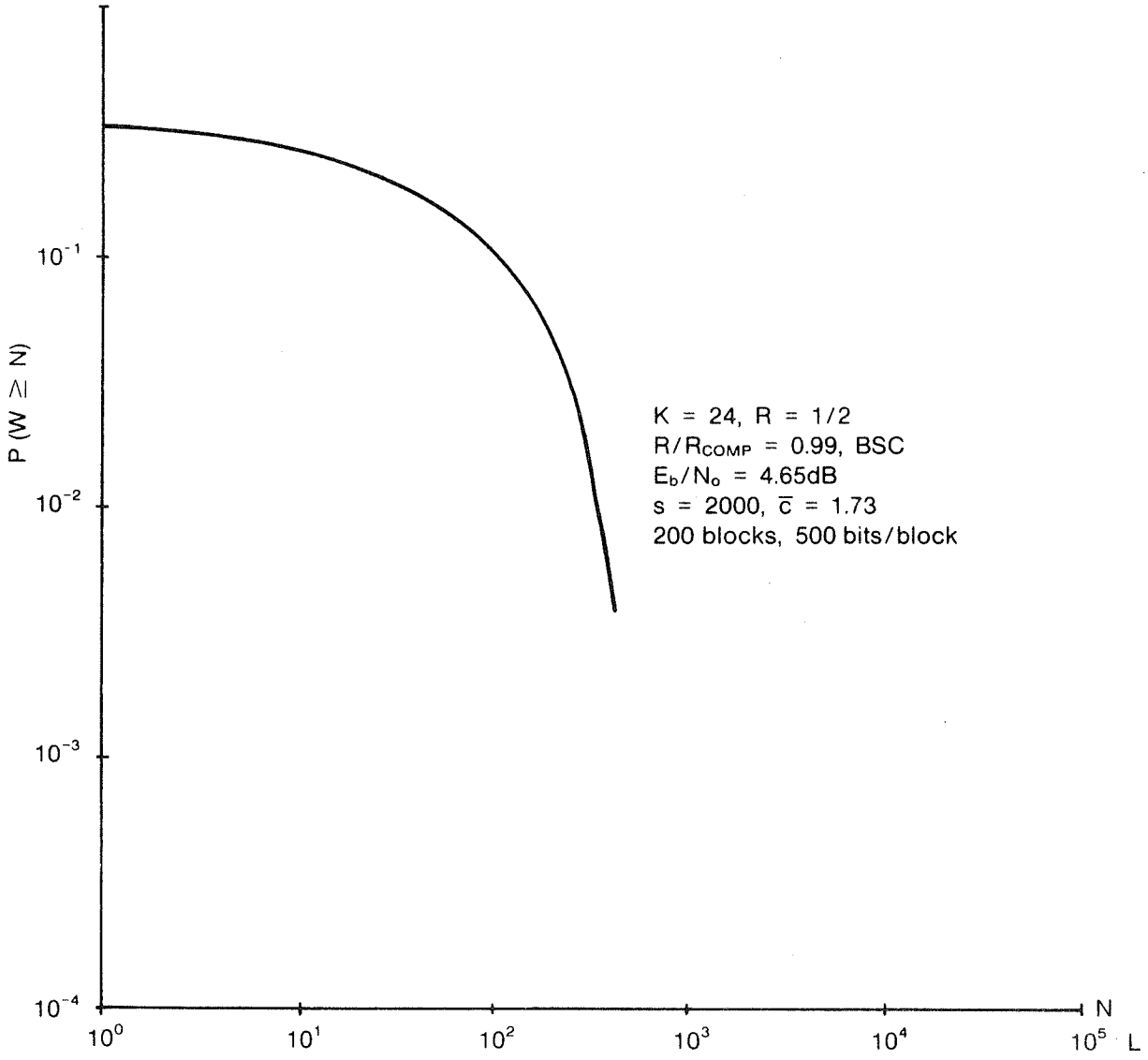

Figure 13: Cumulative de la taille de la file d'attente dans le tampon d'entrée.

Pour la réalisation matérielle de décodeurs séquentiels, un tampon d'entrée ne pouvant contenir que 2 blocs est une solution fort intéressante. L'analyse ci-dessus montre qu'on peut échanger taille de la pile et gain de vitesse pour atteindre un compromis adéquat pour une probabilité de débordement donnée. Un choix judicieux entre la taille de la pile, la longueur des blocs et le gain de vitesse doit donc être effectué dans une réalisation matérielle.

Des résultats de simulations avec l'algorithme Z-J et ses variantes sont donnés au Tableau 3 pour des piles de taille 10000 noeuds, des blocs de longueur 523 bits, $R/R_{comp} = 0.99$ et des valeurs de β variant de 3.19 à 0.96. On voit que pour des gains de vitesse faibles, ($G = 3$) et un β élevé ($\beta = 3.19$) la taille moyenne de la file d'attente augmente avec le nombre d'extensions simultanées de l'algorithme (ou encore avec le nombre moyen de calculs par bit). Cependant les tailles de tampons correspondant à des probabilités de débordement de 10^{-2} et 10^{-3} sont sensiblement les mêmes pour tous les algorithmes, indiquant encore une réduction de la variabilité de l'effort de calcul pour un effort moyen plus grand.

Comme on pouvait le prévoir, les valeurs moyennes de la file d'attente et les tailles du tampon correspondant à des probabilités de débordement de 1×10^{-2} et 1×10^{-3} diminuent toutes avec une augmentation du gain de vitesse. En particulier por $G = 10$ et $\beta = 0.96$, les valeurs moyennes tombent à 4 ou 5 bits et les faibles valeurs de la taille du tampon d'entrée correspondant à une probabilité de débordement de 10^{-3} peuvent laisser croire qu'effectivement la file d'attente n'atteindra jamais une longueur de $2 L = 1046$ bits. D'ailleurs au cours de simulations extensives cette valeur de la file d'attente n'a jamais été atteinte.

Le tableau 3 donne aussi le pourcentage du temps où le décodeur est actif, c'est-à-dire n'a pas un tampon d'entrée vide et exécute l'une ou l'autre des opérations de l'algorithme. Ce pourcentage apparaît dans les

rangées % ACTIF. L'influence du gain de vitesse du décodeur G se reflète bien dans ce pourcentage qui, pour l'algorithme Z-J, par exemple, varie de 55% à 16% lorsque le gain de vitesse varie de 3 à 10. On peut voir aussi qu'au gain de vitesse égal à 3, l'algorithme 7 - Sous-Piles fait fonctionner le décodeur très près de sa capacité de calcul. Le gain de vitesse a donc un impact direct sur le type de variante que l'on peut utiliser. À première vue ces résultats semblent être nettement différents des résultats classiques du décodage séquentiel, où la fonction de répartition de la file d'attente au tampon d'entrée suit une loi asymptotiquement Pareto [6]-[12]. Cependant ces résultats classiques antérieurs ont été obtenus avec l'algorithme de Fano [4] qui ne comporte pas de pile mais seulement un tampon d'entrée. L'absence de pile de l'algorithme de Fano correspond à une pile qui ne peut jamais déborder pour l'algorithme Z-J, c.a.d. à une pile de taille infinie. Ceci revient donc à avoir un paramètre β infiniment grand et comme on l'a vu plus haut (Eq. (16)) à une taille de file d'attente non bornée. Il n'y a donc aucune ambiguité avec les résultats classiques de l'algorithme de Fano, mais comme ici les débordements peuvent survenir aussi bien dans la pile que dans le tampon d'entrée, on peut donc comme on le verra plus loin, contrôler avec avantage ces débordements pour qu'ils apparaissent seulement dans la pile.

TABLEAU 3: FILE D'ATTENTE AU TAMPON D'ENTRÉE

R/R_{comp} = 0.99, Eb/No = 4.64 dB, K = 24, R = ½

200 blocs simulés, 523 bits/bloc, pile S = 10,000

(i) Gain de vitesse du décodeur G = 3, β = 3.19

ALGORITHME / PARAMÈTRE	Z-J	2-CHEMINS	N SOUS-PILES		ADAPTATIF
			N = 5	N = 7	M(D) = 1+½ D
% ACTIF	55	81	72	95	73
FILE MOYENNE $\bar{W}$	87	170	134	488	138
TAMPON MIN. $B_{min}^{(2)}$	1120	1120	1260	1260	1260
TAMPON MIN. $B_{min}^{(3)}$	1410	1260	1410	1580	1410

(ii) Gain de vitesse du décodeur G = 5, β = 1.91

ALGORITHME / PARAMÈTRE	Z-J	2-CHEMINS	N SOUS-PILES		ADAPTATIF
			N = 5	N = 7	M(D) = 1+½ D
% ACTIF	33	48	43	57	44
FILE MOYENNE $\bar{W}$	21	25	23	30	24
*TAMPON MIN. $B_{min}^{(2)}$	560	500	500	500	560
*TAMPON MIN. $B_{min}^{(3)}$	800	800	800	800	800

(iii) Gain de vitesse du décodeur G = 10, β = 0.96

ALGORITHME / PARAMÈTRE	Z-J	2-CHEMINS	N SOUS-PILES		ADAPTATIF
			N = 5	N = 7	M(D) = 1+½ D
% ACTIF	16	24	22	28	22
FILE MOYENNE $\bar{W}$	5	5	4	4	4
*TAMPON MIN. $B_{min}^{(2)}$	125	125	125	100	125
*TAMPON MIN. $B_{min}^{(3)}$	355	355	355	280	355

Note * TAMPON MIN. $B_{min}^{(k)} = B$: $Pr(W \geq B) = 10^{-k}$, k = 2,3

6. PROCÉDURE DE RETRANSMISSION

Dans certaines liaisons telles les communications entre ordinateurs, une probabilité d'erreur inférieure à 10^{-10} est souvent requise. Dans ces conditions même le décodage séquentiel s'avère insuffisant et une procédure de retransmission devient nécessaire. Donc, en principe les blocs provoquant un débordement de la pile ou du tampon devraient être retransmis. Il s'agit donc de déterminer si ces débordements devraient se produire dans la pile ou le tampon d'entrée.

Le choix d'un tampon de grande taille entraîne nécessairement l'utilisation d'un grande pile, alors qu'un débordement de la pile oblige le décodeur à expurger du tampon le bloc qui a fait déborder la pile et le force à décoder un nouveau bloc. Si le bruit accompagnant ce nouveau bloc et les blocs suivants est normal, alors ces blocs seront facilement décodés, permettant au décodeur de réduire, voire même éliminer la file d'attente qui aurait pu s'accumuler dans le tampon d'entrée. Il apparaît donc déjà préférable de confiner les débordements dans la pile plutôt que dans le tampon. De plus comme la distribution du remplissage de la pile est de type Pareto, un grand accroissement de la taille de la pile n'a que peu d'effets sur sa probabilité de débordement. Enfin, d'un point de vue pratique chaque entrée d'un noeud dans la pile requiert quelque 75 à 100 bits alors qu'une entrée dans le tampon d'entrée ne nécessite que 2 à 6 bits, de sorte que toute réduction de la taille de la pile peut se traduire par une réduction de l'espace mémoire et une simplification substantielles du décodeur.

Par conséquent il s'avère avantageux d'utiliser une pile de taille modeste afin d'y détecter très tôt par débordement les blocs difficiles à décoder, et d'en demander leur retransmission. Ces blocs étant expurgés assez tôt, le tampon d'entrée devient peu susceptible à des débordements et donc sa taille peut être réduite sans grande conséquence. Comme il a été montré plus haut, en choisissant le gain de vitesse G tel que $G \geq (S/2L) \geq \overline{C}$, un tampon de taille (2L) branches ne débordera jamais. La taille minimum de la pile S pourra donc être déterminée en fonction de la longueur des blocs.

Un autre avantage à retransmettre les blocs qui débordent provient du fait que les blocs nécessitant un très grand effort de calcul sont souvent décodés en erreur. En retransmettant ces blocs, ces erreurs peuvent donc être évitées. Par conséquent une telle procédure de retransmission permet de réduire au minimum la taille du tampon, et de réduire la probabilité d'erreurs non détectées, au prix d'une faible détérioration du taux de codage effectif. Ainsi le décodeur séquentiel servirait à la correction et à la détection des erreurs.

La technique de retransmission, a été simulée sur ordinateur utilisant la procédure ARQ sélectif [3], c'est-à-dire seuls les blocs causant un débordement sont retransmis (voir Figure 14). Utilisant encore un code de taux R = ½ et de longueur de contrainte K = 24, et des blocs de longueur L = 523 bits, on a fait varier le gain de vitesse G du décodeur de 2 à 20 et la taille de la pile S de 1000 à 10,000 entrées. Sauf exceptions le nombre de blocs à transmettre est égal à 200 avec des valeurs de E_b/N_o dans le canal direct égale à 4.64 dB et 4.89 dB, correspondant à R/R_{comp} = 0.99 et 0.95 respectivement. Le canal de retour est considéré être sans bruit.

Pour chaque taille de la pile les 200 blocs sont d'abord transmis dans un premier cycle. Tous les blocs ayant débordés sont ensuite retransmis dans un 2e cycle. Si au cours de ce 2e cycle des blocs débordent encore, ils sont retransmis dans un 3e cycle et ainsi de suite jusqu'au décodage complet des 200 blocs.

À la fin de la simulation la taille maximum de la file d'attente W_{max} est observée ainsi que le nombre de blocs retransmis, le nombre de cycles de retransmissions, le nombre d'erreurs, le nombre moyen de calculs par bit, et le taux effectif de codage ou "throughput" (où l'effet de la queue est pris en considération). Un exemple de résultats est fourni au Tableau 4 où la taille de la pile varie de 1,300 à 10,000 entrées. Comme on pouvait le prévoir, une pile de petite taille donne lieu à de nombreuses retransmissions et même plusieurs cycles de retransmissions. La taille de

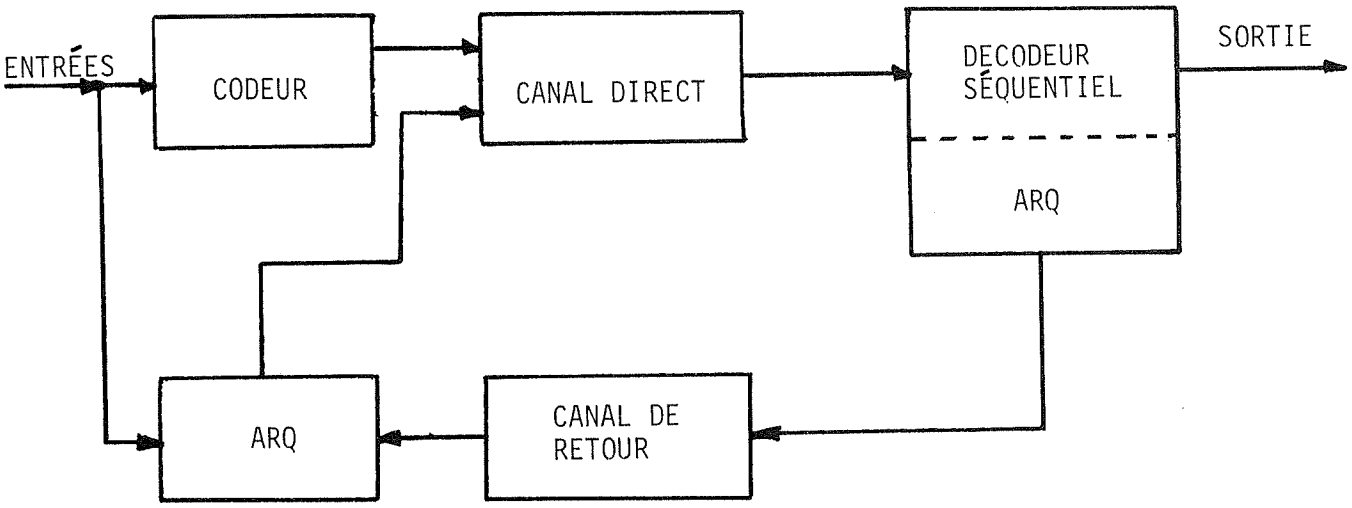

Figure 14: Schéma de principe du décodeur séquentiel avec ARQ

TABLEAU 4: DÉCODAGE SÉQUENTIEL AVEC RETRANSMISSIONS PAR DÉBORDEMENT DE LA PILE

R/R_{comp} = 0.95, Eb/No = 4.89 dB, K = 24, R = ½

L = 523 bits/bloc; Quantification ferme.

PILE (ENTRÉES)	CALCUL MOYEN $\overline{C}$	TRAVAIL MOYEN $\overline{T}$	THROUGHPUT R	DÉBORDEMENTS NOMBRE BLOCS	CYCLES DE RETRANSMISSION	ERREURS
1300	1.73	2.19	0.329	91/200	4	0
1500	1.47	1.86	0.400	39/200	3	0
2000	1.37	1.74	0.447	14/200	1	0
2500	1.37	1.76	0.455	10/200	1	0
5000	1.44	1.89	0.469	6/300	1	22
10000	1.49	1.99	0.475	3/500	1	67

la pile augmentant, le nombre de retransmissions diminue, et très vite un seul cycle de retransmissions suffit. Sur le Tableau 4, on peut voir qu'au delà d'une pile de taille 2,000 entrées tous les débordements sont résolus en un seul cycle de retransmission, et pour une pile de 10,000 entrées seulement 3 blocs sur 500 ont débordé. Les retransmissions nécessitant un effort de calcul supplémentaire, le nombre moyen de calculs $\bar{C}$ a tendance à augmenter à mesure que la taille de la pile diminue, passant de 1.71 pour une pile de 1,300 entrées à 1.49 pour une pile de 10,000 entrées. Naturellement retransmissions et effort de calcul se répercutent sur le "throughput" qui passe de 0.366 pour une pile de 1,300 entrées à 0.475 pour une pile de 10,000 entrées. (Tenant compte de la queue, le throughput maximum est égal à 0.478).

Tel que mentionné plus haut, les blocs difficiles à décoder sont souvent décodés en erreur. Par conséquent une pile de très grande taille permettant de gros efforts de calculs pour décoder un bloc peut délivrer ce bloc entaché d'erreurs, alors qu'une petite pile aura vite débordé et nécessite une retransmission de ce bloc. Le Tableau 4 illustre bien ce phénomène où une pile de 5,000 entrées a délivré 22 erreurs sur 300 blocs et une pile de 10,000 entrées a délivré 67 erreurs sur 500 blocs. Par contre aucune erreur n'a été observée pour des piles de tailles comprises entre 1,300 et 2,500 entrées.

Un examen détaillé des opérations de décodage a montré que les temps d'exécution d'un calcul n'étaient pas les mêmes pour une extension de chemin en avant dans l'arbre et pour des recherches arrières. En recherche arrière la détermination du noeud à prolonger à partir de la meilleure sous-pile vide pouvant nécessiter plusieurs essais successifs, le temps d'un calcul est plus long que lors d'une extension avant où la détermination du noeud à prolonger est en général immédiate. La mesure des opérations de

calcul en cours des simulations a montré qu'en moyenne le temps d'un calcul en recherche arrière est environ 1.5 fois plus long que celui d'une extension avant. On peut donc évaluer le "travail moyen" du décodeur par

$$\overline{T} = \overline{C}_{AV} + 1.5\,\overline{C}_{AR} \qquad (22)$$

où $\overline{C}_{AV}$ est le nombre moyen de calculs correspondant aux extensions avant, et où $\overline{C}_{AR}$ est le nombre moyen de calculs correspondant aux recherches arrières. Le Tableau 4 donne les valeurs de travail moyen lorsque la pile varie de 1,300 à 10,000. Le travail moyen passe par un minimum pour les piles de taille 2,000 à 2,500 entrées et croît assez vite pour des piles de très petites tailles. Ce comportement peut s'expliquer par le fait qu'une petite pile aura tendance à déborder très vite dès que le nombre de recherches arrières dépasse un certain seuil. De plus le phénomène tend à se produire assez fréquemment puisque le nombre de retransmissions augmente aussi très vite pour des piles de petites tailles. Le nombre de recherches arrières est donc moins bien compensé par le nombre d'extensions en avant ce qui a tendance à faire augmenter le temps de calcul moyen.

Le Tableau 4 monte que le travail moyen augmente aussi pour les grandes piles (5,000 et 10,000 entrées). Ceci s'explique par le fait qu'avec des piles de très grandes tailles le décodeur peut effectuer de très longues recherches arrières sans débordements, recherches arrières qui auraient provoqué un débordement avec des piles plus petites. Il est bon de remarquer qu'aucune des analyses théoriques de l'effort de calcul du décodeur séquentiel ne fait de distinction entre un calcul en avant et un calcul en recherche arrière, et la notion de travail moyen n'y apparaît donc pas. Cependant dans une réalisation matérielle il faut tenir compte du travail moyen $\overline{T}$ et de sa relation avec la taille de la pile.

Tout comme pour les systèmes sans retransmissions, la fonction de répartition du nombre d'entrées dans la pile suit une tendance Pareto, indépendemment du gain de vitesse du décodeur. La Figure 15 donne les cumulatives du remplissage de la pile pour des tailles de piles variant de 1,300 à 10,000 entrées. On peut voir que les cumulatives sont toutes confondues les unes, avec les autres et que la cumulative du remplissage d'une pile de taille donnée n'est que le prolongement de celle de la taille immédiatement inférieure, démontrant ainsi un comportement remarquablement uniforme.

Comme on l'a vu plus haut, la file d'attente dans le tampon d'entrée dépend de la taille de la pile S, du gain de vitesse G et de la longueur des blocs L. On rappelle que si le paramètre $\beta = \frac{S}{2GL}$ est inférieur à 1 la longueur de la file d'attente W_{max} ne peut dépasser 2L, alors que pour $\beta \geq 1$ la file d'attente peut croître sans limite. Les Figures 16, 17 et 18 donnent les cumulatives des longueurs des files d'attente pour les piles de tailles 1,500, 5,000 et 10,000 entrées respectivement, et pour des gains de vitesses G variant de 2 à 30, ou des β variant de 0.147 à 3.827. Chacune de ces figures montre que pour une pile de taille donnée, toute augmentation du gain de vitesse ou diminution du paramètre β se traduit par une amélioration de la cumulative de la file d'attente. De plus conformément à ce qui a été établi plus haut, on peut voir que pour toutes les valeurs de β inférieures à 1, la file d'attente est restée inférieure à sa longueur maximum W_{max} égale à 2 blocs, soit 2 x 523 = 1046. Cependant cette longueur maximum a été largement dépassée pour tous les cas où β est supérieur à 1. Les Figures 16, 17 et 18 illustrent bien l'interdépendance qui existe entre la taille de la pile, le gain de vitesse et la longueur de la file d'attente, donc la taille du tampon d'entrée requise. Ainsi, pour maintenir β inférieur à 1 et donc se suffire d'un tampon d'entrée de longueur 2L branches, toute augmentation de la taille de la pile entraîne une augmentation proportionnelle de G_{min}, le gain de vitesse minimum du décodeur. Par exemple, ici avec L = 523 branches, G_{min} est respectivement

égal à 1.43, 4.78 et 9.56 pour les piles de tailles S = 1,500, 5,000 et 10,000 entrées. Ainsi, une limitation des débordements de la pile par un choix de piles de grandes tailles entraîne l'exigence de grands gains de vitesse (ce qui en pratique se traduit par une électronique rapide) si on veut limiter la taille du tampon d'entrée à 2L branches. D'autre part, on peut être intéressé à une configuration minimale de décodeur: pile minimum, gain de vitesse minimum et taille de tampon minimum (β 1). Sachant que le gain de vitesse doit être au moins égal à l'effort de calcul moyen $\bar{C}$, la taille minimum de la pile S_{min} est donc

$$S_{min} = 2L\ \bar{C} \qquad (23)$$

avec

$$G_{min} = \bar{C} \qquad (24)$$

Naturellement avec ces valeurs minimum les débordements de la pile seront très fréquents. Se référant au Tableau 4, pour $\bar{C}$ = 1.73 on trouve S_{min} = 1810. Pour cette valeur de $\bar{C}$ la taille de la pile utilisée n'étant que 1,300, il n'est donc pas surprenant que le nombre de blocs ayant débordés soit si élevé: 91 débordements sur 200 blocs à transmettre, nécessitant 4 cycles de retransmissions, et correspondant à environ 31% de blocs ayant débordé. Par comparaison pour $\bar{C}$ = 1.44, S_{min} = 1,506, et donc une pile de taille S = 5,000 ne débordera que très rarement. Ceci est encore bien illustré au Tableau 4 où seulement 6 blocs sur 300 ont débordé, soit une proportion de 2%. En contre partie, tel que mentionné plus haut les gains de vitesse devront être supérieurs à 1.73 et 4.78 si on veut limiter les files d'attente à une longueur maximum de 2L = 1,046 branches.

L'analyse ci-dessus montre que les paramètres clefs d'un décodeur séquentiel, tailles de la pile, du tampon, gains de vitesse et longueurs de bloc sont fortement liés les uns aux autres. Les Figures 19 et 20 montrent l'échange possible qui existe entre le gain de vitesse, la taille de la pile et celle de la file d'attente pour des blocs de longueur fixe L = 523 bits.

Sur chacune des figures la démarcation entre le cas où la file d'attente est bornée à 2L = 1046 branches ($\beta < 1$), et celui où elle est non borné ($\beta \geq 1$) est indiquée par la ligne horizontale d'ordonnée 2L = 1046. Par exemple sur la Figure 20 on peut voir que pour des piles inférieures à 3500 entrées tous les systèmes simulés ont donné une file d'attente bornée. Pour des piles de taille plus grande on voit l'influence du gain de vitesse sur la longueur de la file d'attente. En particulier pour une pile de grande taille (9000 ou 10,000 entrées), un faible gain de vitesse entraîne l'accumulation d'une très grande file d'attente dans le tampon d'entrée, alors que pour des gains de vitesse très élevés, ($G \geq 10$) la taille de la pile n'a presque plus d'influence sur la file d'attente. Dans une application particulière le choix final des paramètres, longueur des blocs, gain de vitesse, taille de pile et taille de tampon devient donc un problème d'ingénierie, et dépendra de cette application et de ses contraintes propres.

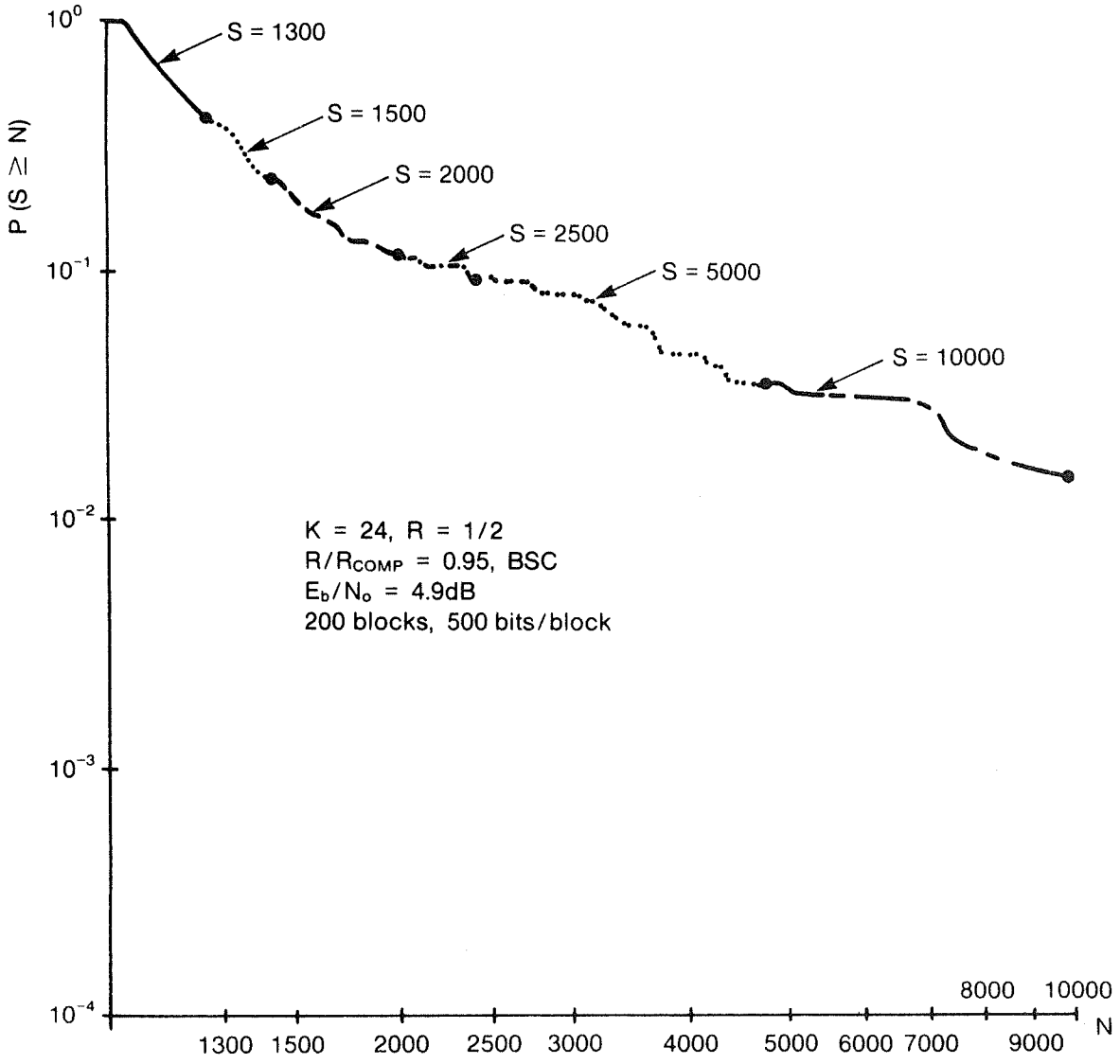

Figure 15: Cumulatives du remplissage de la file, lorsque sa taille varie de S = 1,300 à S = 10,000 entrées.

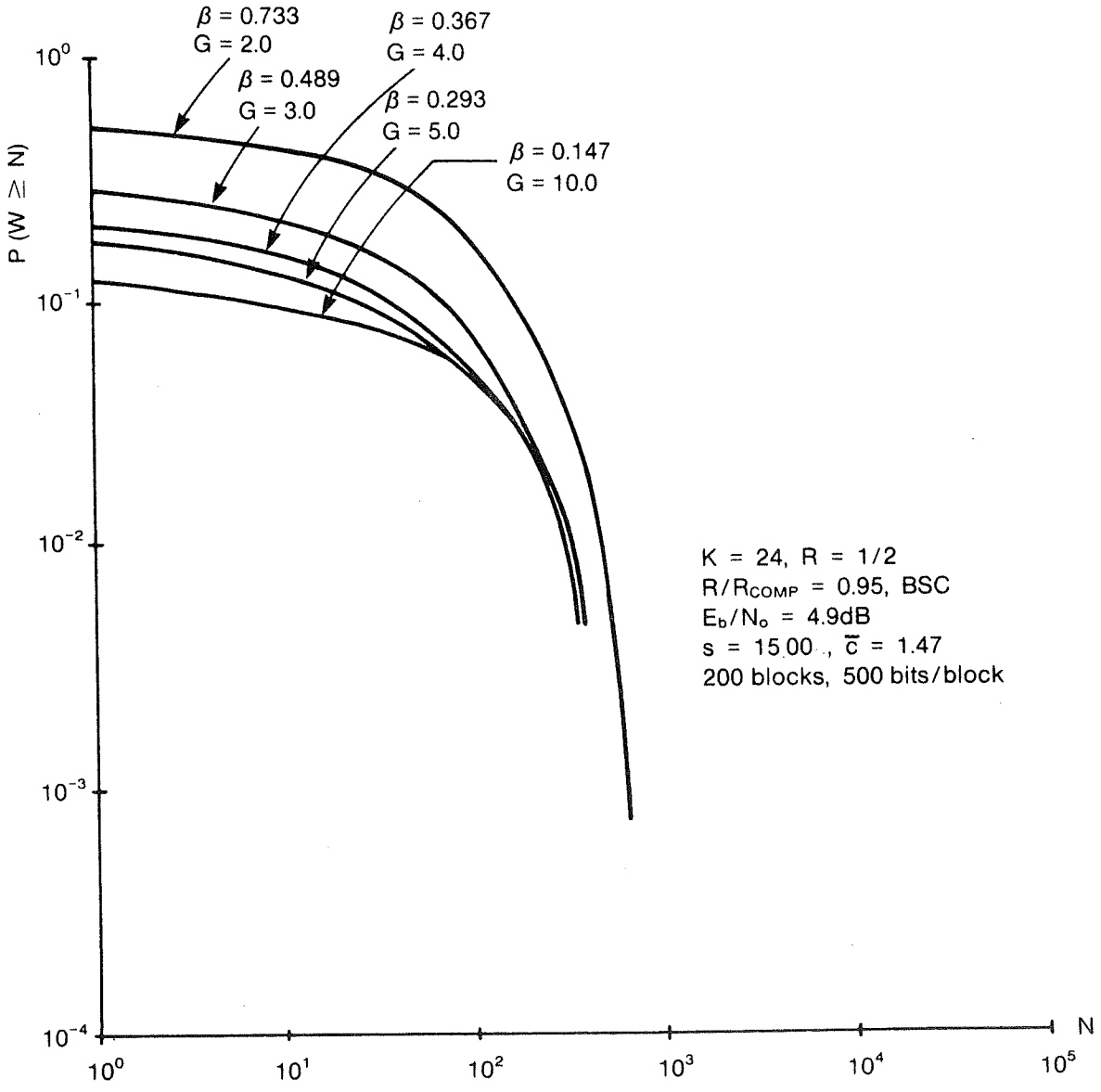

Figure 16: Cumulatives des longueurs des files d'attente dans le tampon d'entrée pour plusieurs gains de vitesse et une pile de taille S = 1,500.

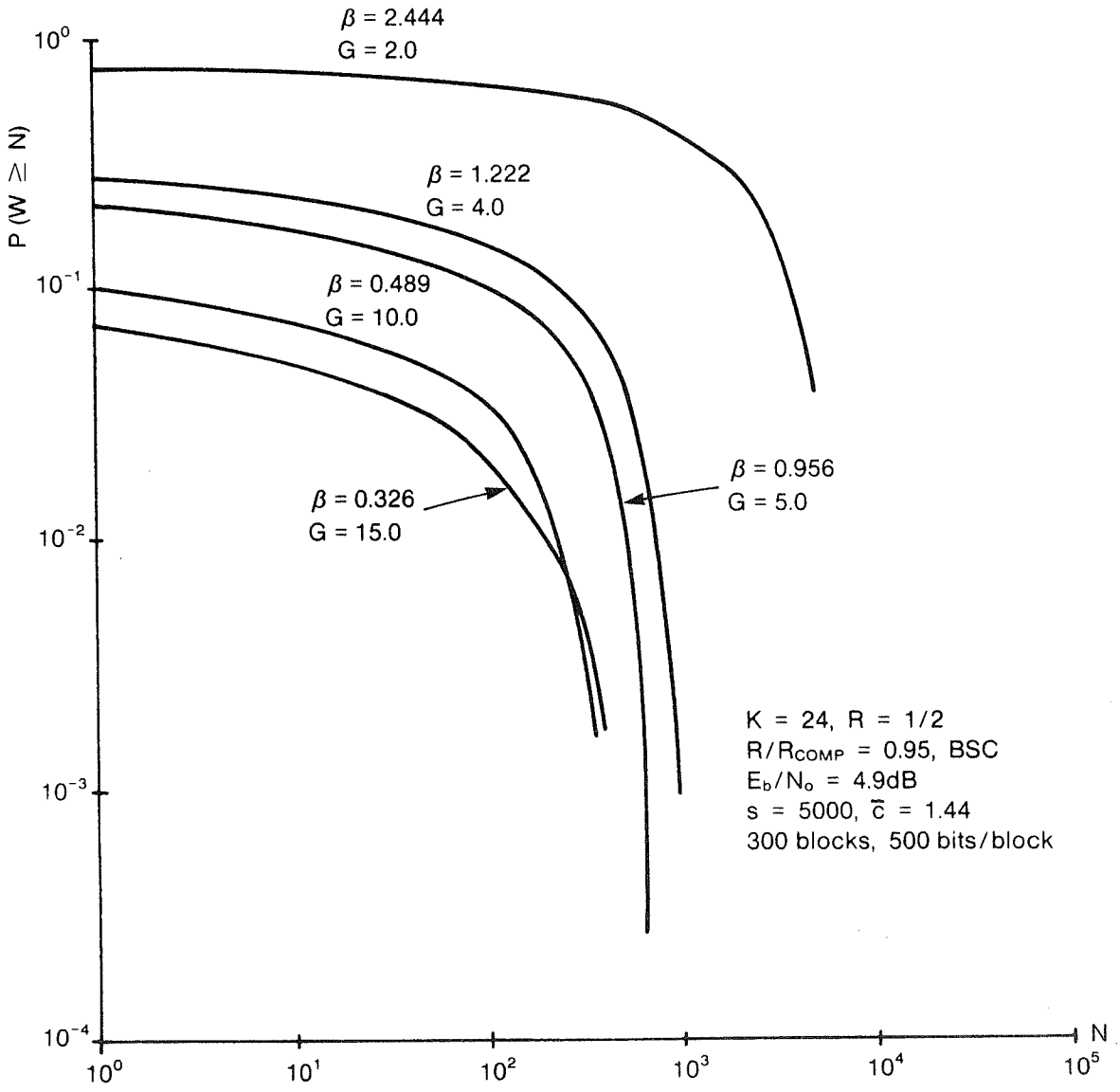

Figure 17: Cumulatives des longueurs des files d'attente dans le tampon d'entrée pour plusieurs gains de vitesse et une pile de taille S = 5,000.

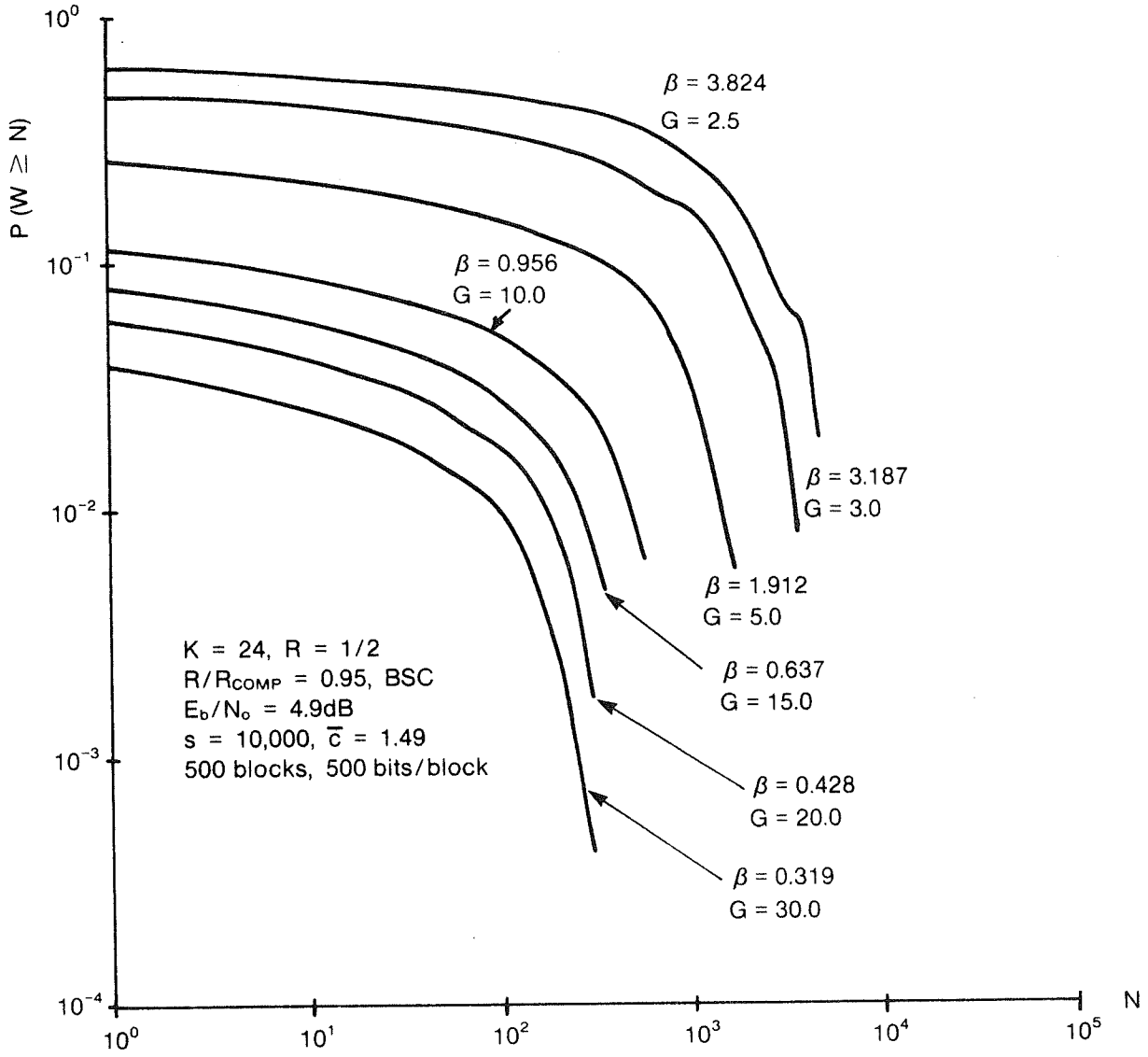

Figure 18: Cumulatives des longueurs des files d'attente dans le tampon d'entrée pour plusieurs gains de vitesse et une pile de taille S = 10,000.

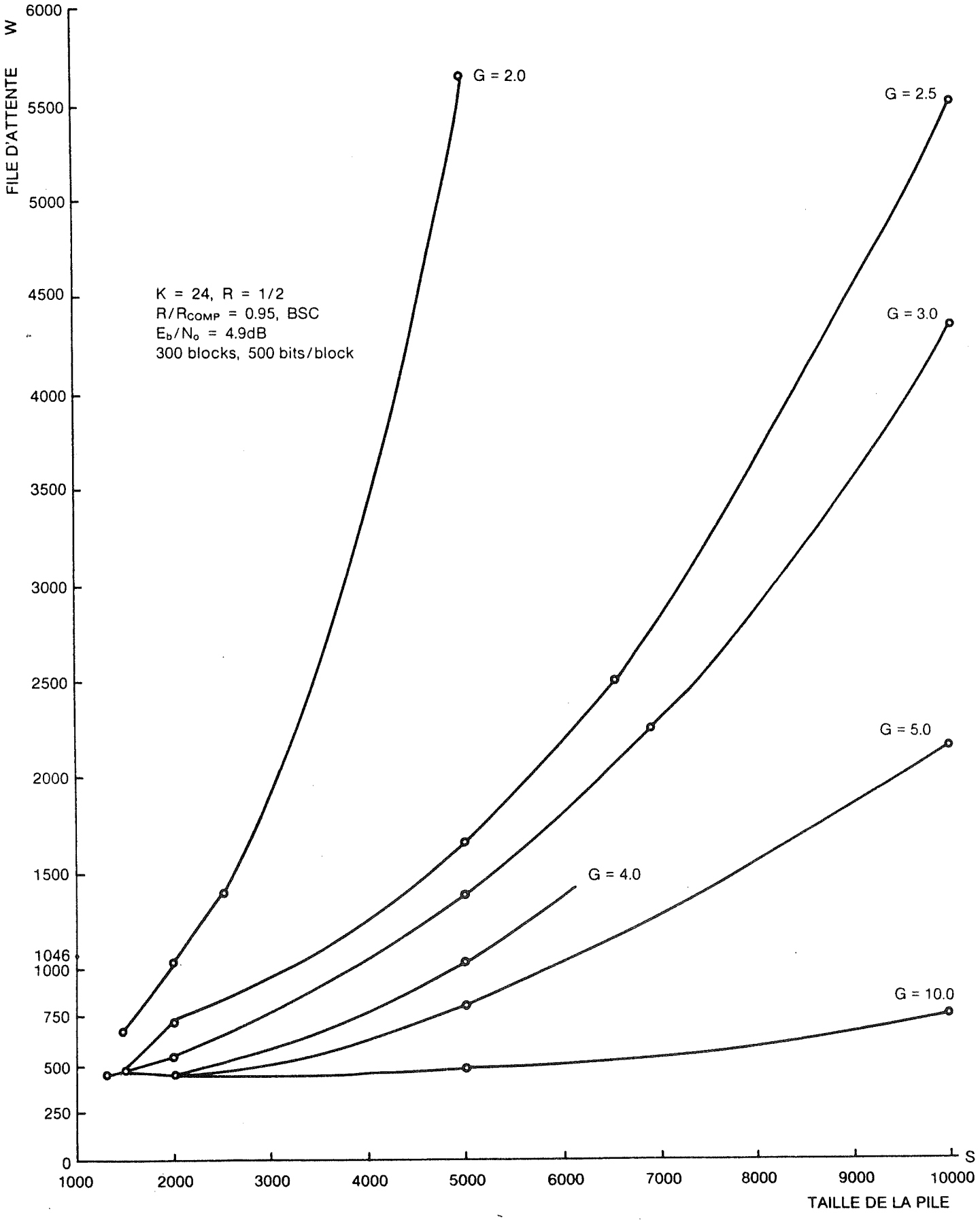

Figure 19: Relations entre la file d'attente et la taille de la pile pour plusieurs gains de vitesse, $R/R_{comp} = 0.95$

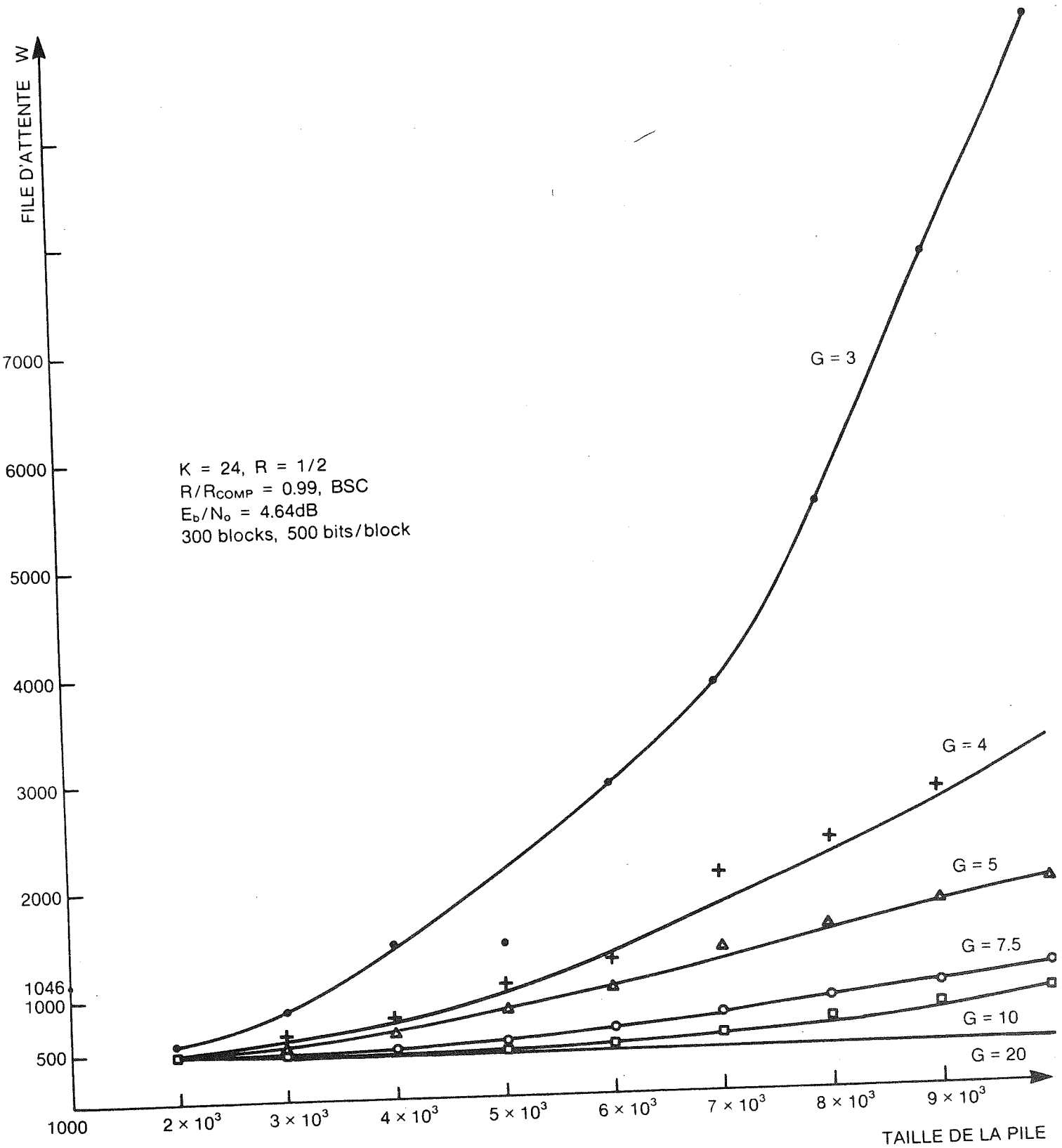

Figure 20: Relations entre la file d'attente et la taille de la pile pour plusieurs gains de vitesse, $R/R_{comp} = 0.99$

7. CONCLUSIONS

Dans cet article nous avons présenté un certain nombre de variantes de l'algorithme de Zigangirov - Jelinek du décodage séquentiel. Toutes ces variantes ont pour objectif la diminution de la variabilité de l'effort de calcul du décodage séquentiel. Les résultats de simulations extensives ont démontré que cette variabilité de l'effort de calcul peut être grandement réduite au coût d'un accroissement de l'effort de calcul moyen, mais aussi sans dégradation de la performance d'erreur. La fonction de répartition du remplissage de la pile dans toutes ces variantes étant le reflet de l'effort de calcul, cette fonction est toujours de type Pareto, avec un exposant Pareto qui croît avec le nombre moyen de calculs effectués. Le choix de l'algorithme à utiliser dépendra de la taille de la pile disponible. La taille de la pile limitant l'effort de calcul maximum par bloc, une analyse simple de la file d'attente dans le tampon d'entrée a montré qu'avec un gain de vitesse suffisant on peut limiter la taille du tampon d'entrée à 2 longueurs de bloc sans débordement. Enfin en contrôlant les débordements de la pile et en utilisant une procédure de retransmission des blocs ayant débordé, on peut contrôler la probabilité de débordement du tampon et réduire sensiblement la probabilité d'erreur au coût d'une faible réduction du taux de codage effectif. Pour finir on a montré qu'il existe un échange possible entre la taille de la pile, la longueur des blocs, la taille du tampon et le gain de vitesse du décodeur, et qu'un choix judicieux de la taille de la pile et du gain de vitesse ne fera jamais déborder un tampon de taille deux longueurs de bloc. Ces résultats pourront être avantageusement mis à profit dans une réalisation matérielle de décodeur pour réduire substantiellement l'espace mémoire repris, et faire du décodeur séquentiel à pile une possibilité pratiquement attrayante pour la correction des erreurs dans des voies de communication très bruitées.

RÉFÉRENCES

[1] F. JELINEK, "A Sequential Decoding Algorithm Using a Stack", IBM Journal of Research and Development, Vol. 13, Nov. 1969.

[2] A.J. VITERBI, "Convolutional Codes and Their Performance in Communication Systems", IEEE Trans. on Com. Tech., Vol. COM-19, Oct. 1971.

[3] V. BHARGAVA, D. HACCOUN, R. MATYAS, P. NUSPL, "Digital Communications by Satellite", John Wiley, New York, Oct. 1981.

[4] R.M. FANO, "A Heuristic Discussion of Probabilistic Decoding", IEEE Trans. on Inform. Theory, Vol. IT-9, pp. 64-73, April 1963.

[5] R.G. GALLAGER, "Information Theory and Reliable Communication", John Wiley, New York, 1968.

[6] I.M. JACOBS et E.R. BERLEKAMP, "A Lower Bound to the Distribution of Computation for Sequential Decoding", IEEE Trans. on Information Theory, Vol. IT-13, April 1967, pp. 167-174.

[7] D. HACCOUN, M. FERGUSON, "Generalized Stack Algorithm for the Decoding of Convolutional Codes", IEEE Trans. on Inf. Theory, Vol. IT-21, Nov. 1975, pp. 638-651.

[8] P.R. CHEVILLAT, D.J. COSTELLO, "A Multiple Stack Algorithm for Erasure-free Decoding of Convolutional Codes", IEEE Trans. on Comm., Vol. COM-25, Dec. 1977, pp. 1460-1470.

[9] D. HACCOUN, "Décodeur séquentiel haute vitesse; réalisation et tests préliminaires", Rapport Technique No EP 84-R-19, Ecole Polytechnique de Montréal, juin 1984, 88 pages.

[10] D. HACCOUN, M. DUFOUR, "Stack and Input Buffers Overflows of Stack Decoding Algorithms", Book of Abstracts, IEEE International Symposium on Information Theory, Santa Monica, California, Feb. 1981.

[11] D. HACCOUN, "Problèmes de débordement de décodeurs séquentiels à pile", 8e Colloque GRETSI sur le traitement du signal et ses applications, Nice, France, juin 1981, pp. 919-923.

[12] J.M. WOZENCRAFT and I.M. JACOBS, "Principles of Communications Engineering", J. Wiley, New York, 1965.

[13] D. HACCOUN, "A Branching Process Analysis of the Average Number of Computations of the Stack Algorithm", IEEE Trans. on Information Theory, Vol. IT-30, No. 3, May 1984, pp. 497-508.

[14] P.Y. PAU, D. HACCOUN, "Sequential Decoding with ARQ", Book of Abstracts, Proc. IEEE International Symposium on Information Theory, St-Jovite, Québec, Sept. 1983.

[15] A. DRUKAREV, D.J. COSTELLO Jr., "Hybrid ARQ Error Control Using Sequential Decoding", IEEE Trans. Inform. Theory, Vol. IT-29, July 1983, pp. 521-535.

[16] A. JANELLE, D. HACCOUN, "Décodage adaptatif des codes convolutionnels", Rapport Technique EP 78-R-18, Ecole Polytechnique de Montréal, 1978, 156 pages.

[17] R. JOHANNESSON, "Robustly Optimal Rate One-half Binary Convolutional Codes", IEEE Trans. on Inform. Theory, July 1975, pp. 464-468.