Predicția salturilor

Predicția salturilor este o cerință importantă în sistemele de calcul ce utilizează intens tehnica pipe-line. Prin predicția salturilor se optimizează utilizarea structurii pipe-line evitîndu-se reinițializarea acesteia dacă intrucțiunile corespunzătoare saltului în program au fost încărcate în mod eronat.

În situația unei instrucțiuni de salt condiționat instrucțiunea țintă nu poate fi încarcată în pipe-line decît după ce s-a calculat adresa de salt şi nu s-a evaluat condiția de salt. Pentru salturile necondiționate trebuie calculată numai adresa de salt.

Pînă cînd aceste informații sînt disponibile structura pipe-line aşteaptă sau încarcă o instrucțiune tintă posibilă; în momentul cînd informațiile sînt disponibile se poate decide o reinițializare a structurii (în situația în care în pipe-line s-au încărcat instrucțiuni țintă în mod eronat). În ambele situații apare o degradare a performanței structurii pipe-line. Utilizarea predicției salturilor conduce la o atenuare a acestei degradări de performanță datoritătaptului că în majoritatea cazurilor instrucțiunile țintă sînt încărcate correct.

Instrucțiunile de salt se impart în două categorii:

- salturi statice (care se regăsesc în codul binar şi sînt cunoscute înainte de execuția programului)
- salturi dinamice (care apar în urma execuției şi nu sînt cunoscute înainte de execuție)

Predicția salturilor dinamice este mai dificilă decît predicția salturilor statice.

Ideea de bază a predicției salturilor este memorarea istoriei fiecărui salt (dacă s-a efectuat sau nu s-a efectuat) şi luarea deciziei (salt efectuat / salt ne-efectuat) pe baza acestei istorii. Se definesc anumite tipare (*pattern*) de diferite lungimi care indică în timp dacă saltul; s-a efectuat sau nu. Aceste tipare sînt dependente de tipul de program care se execută.

Există mai multe tipuri de programe (*task-uri*):

T1 – procesarea bazelor de date
T2 – programe de căutare, editare, compilare şi testare
T3 – programe de rezervare hotelieră, tranzacții bancare
T4 – programe utilitare pentru manevrarea de date

În tabelul 1 sînt prezentate numărul de isnstrucțiuni de salt pentru fiecare tip de task:

	T1	T2	T3	T4
Numărul total de instrucțiuni	1,300,881	1,325,359	1,309,178	1,667,468
Numărul salturilor dinamice	285,528	321,441	312,865	359,550
Numărul salturilor statice	19,176	27,878	21,202	15,491

Predicția salturilor utilizează un buffer de memorie (BTB – *Branch Target Buffer*) care conține adresele instrucțiunilor țintă pentru fiecare salt precum şi informația necesară predicției. Bufferul BTB este adresat cu ajutorul adresei instrucțiunii de salt. Deoarece în mod evident nu se poate utilize un BTB excesiv de mare se vor utilize tehnici de mapare a adresei instrucțiunii de salt în spațiul de adresabilitate al BTB (tehnici similare mapării memoriei cache). Dimensiunea bufferului BTB influențează rata de predicție (figura 1).

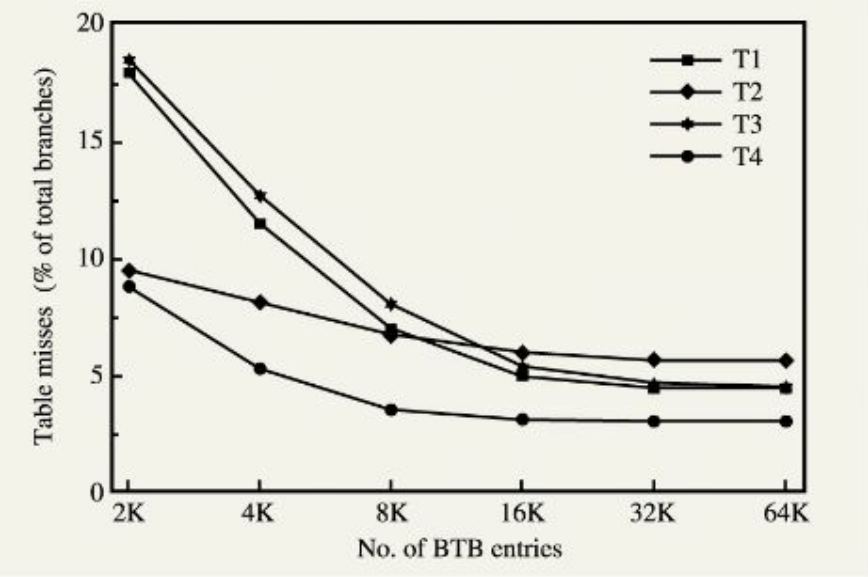


Figura 1. Variația ratei de predicție funcție de dimensiunea BTB

Dimensiunea mare a BTB poate fi în aparență un avantaj. Totuşi un BTB mare are următoarele dezavantaje:

- se reduce din dimensiunea memoriei sistemului
- se stochează adrese effective (nu adrese fizice) ceea însemnă că la comutarea task-urilor tot conținutul BTB este inutil sau chiar contraproductiv

Metoda de mapare a BTB (similară memoriei cache) influențează rata de predicție ca în figura 2.

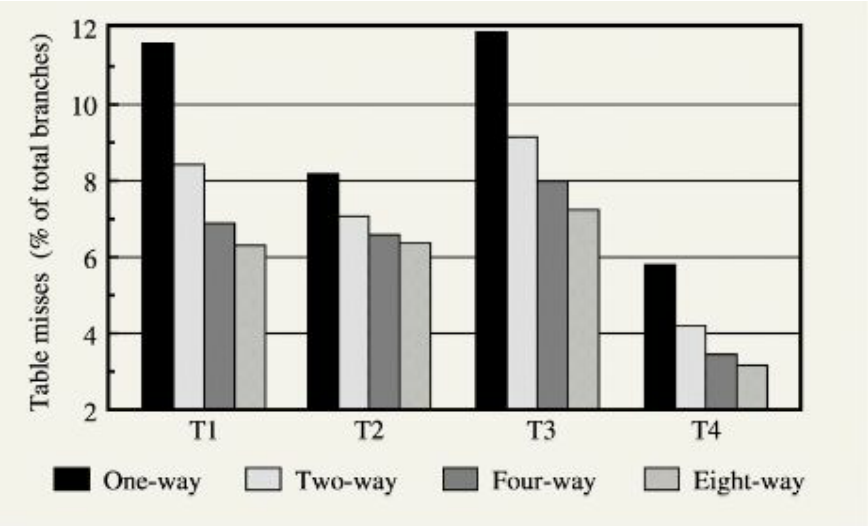


Figura 2. Variația ratei de predicție funcție de metoda de mapare a BTB

În situaţia în care procesorul încarcă mai mult de o singură instrucţiune într-un ciclu atunci bufferul BTB trebuie accesat pe blocuri de date. Rata de predicţie va scadea uşor ca în figura 3.

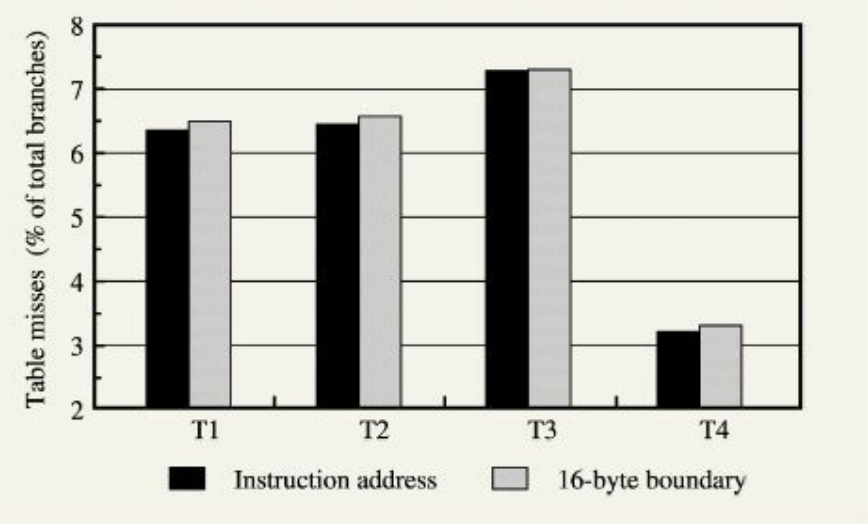


Figura 3. Variaţia ratei de predicţie funcţie de modul de adresare a BTB

Predicţia direcţiei salturilor

Salturile condiţionate sînt prezise utilizînd informaţia anterioară despre modul cum s-au efectuat aceste salturi.

Cea mai simplă cale de predicţie a saltului este aceea de a presupune că un nou salt se va efectua în aceeaşi direcţie ca salturile anterioare. O asemenea predicţie se numeşte predicţie locală cu o istorie de 1 bit (*local prediction 1 bit history*). Evident metoda este mai bună dacă istoria are mai mulţi biţi care memorează direcţia saltului (1 se efectuează saltul, 0 nu se efectuează saltul). Tabelul 1 ilustrează modul de predicţie a salturilor pentru un tipar de 3 biţi.

Tabelul 1

Tiparul (*pattern*)	**Predicţie**	**Saltul urmator efectuat este efectuat(%)**
000	ne-efectuat (0)	7.8
001	ne-efectuat (0)	34.1
010	efectuat (1)	51.9
011	efectuat (1)	67.9
100	ne-efectuat (0)	32.6
101	efectuat (1)	64.4
110	efectuat (1)	79.1
111	efectuat (1)	97.7

Se utilizează un registru de deplasare (cu n biţi pentru un predictor local cu istorie de n biţi). Acest registru de deplasare conţine ultimele *n* decizii. Cel mai din stînga bit reprezintă

decizia cea mai veche. Construirea istoriei presupune ca bucla să se fi executat de un anumit număr de ori timp în care saltul nu este predictibil).

Istoria fiecărui salt este stocată în bufferul BTB.

Analizele pe o gamă variată de programe indică faptul că dacă numărul de biți din tipar creşte rata de predicție creşte. Totuşi prin adăugarea unui bit (de la o istorie cu 2 biți la o istorie cu 3 biți) creşterea nu este semnificativă.

Se poate utiliza, pentru fiecare salt, un contor de 2 biți care este incrementat ori de cite ori saltul se efectuează şi este decrementat ori de cite ori saltul nu se efectuează. Operațiile de incrementare, respectiv decrementare, se efectuează cu saturare astfel încît contorul ia valorile 0,1,2 sau 3. Predicția se efectuează astfel : dacă valoare contorului este 0 sau 1 – saltul nu se efectuează, dacă valoarea contorului este 2 sau 3 – saltul se efectuează. Această schemă de predicție poate fi asimilată unui automat cu 4 stări ca în figura 4.

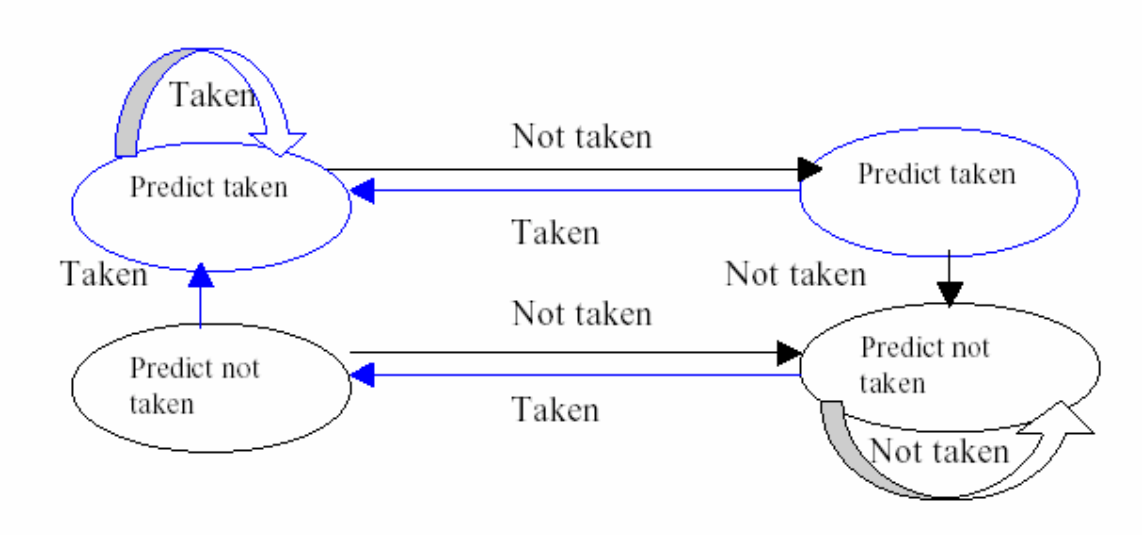


Figura 4. Schema de predicție cu contor de 2 biți cu saturare

Rezultatele experimentale obținute în literatură pentru un predictor local cu istorie de 3 biți şi pentru un predictor cu contor de 2 biți cu saturare sînt ilustrate în figurile 5 şi 6.

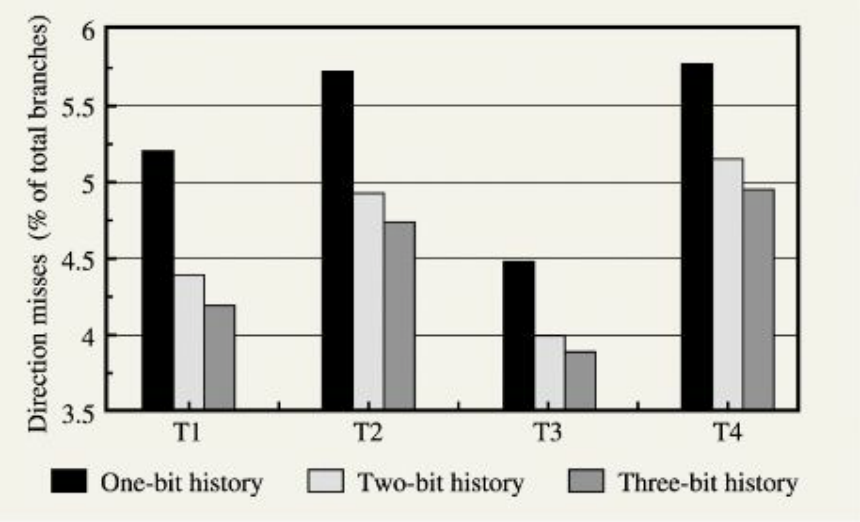


Figura 5. Predicția cu un predictor local cu istorie de 3 biți

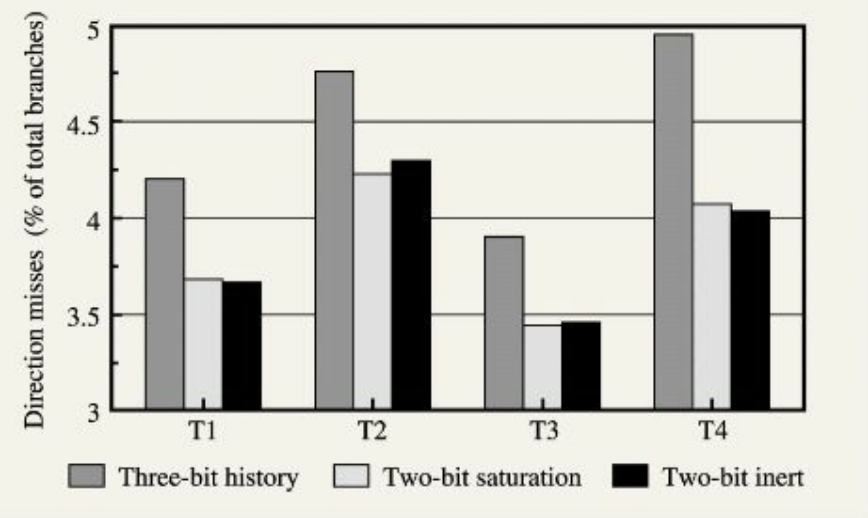


Figura 6. Predicția cu un predictor cu contor de 2 biți cu saturare

O schemă de predicție a salturilor mai eficientă este schema de predicție adaptivă cu 2 niveluri. Această schemă presupune existența a 2 tabele: o tabelă BTB şi o tabelă de contori de 2 biți cu saturare. Tabela BTB conține istoria pe *n* biți a fiecărui salt; conținutul BTB adresează tabela de contori; decizia se ia ca în cazul schemei de predicn cazul schemei de predicție cu contor cu saturare; se actualizează contorii şi tabela BTB. Schema de predicție adaptivă cu 2 niveluri este ilustrată în figura 7.

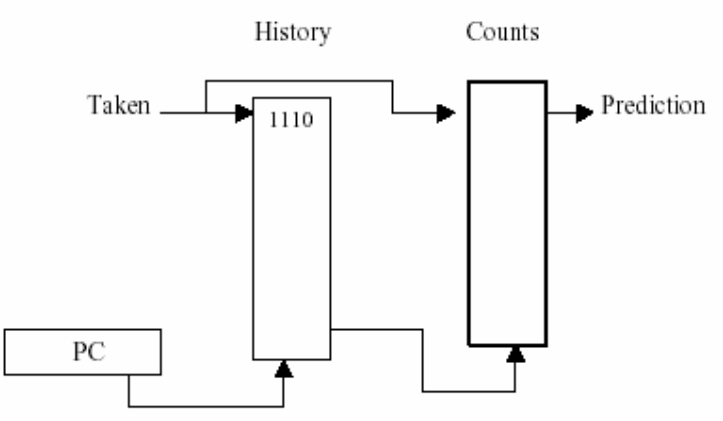


Figura 7. Predicția adaptivă cu 2 niveluri

În figura 8 este ilustrată rata de predicție pentru un predictor adaptive cu 2 niveluri funcție de numărul de biți de adresă.

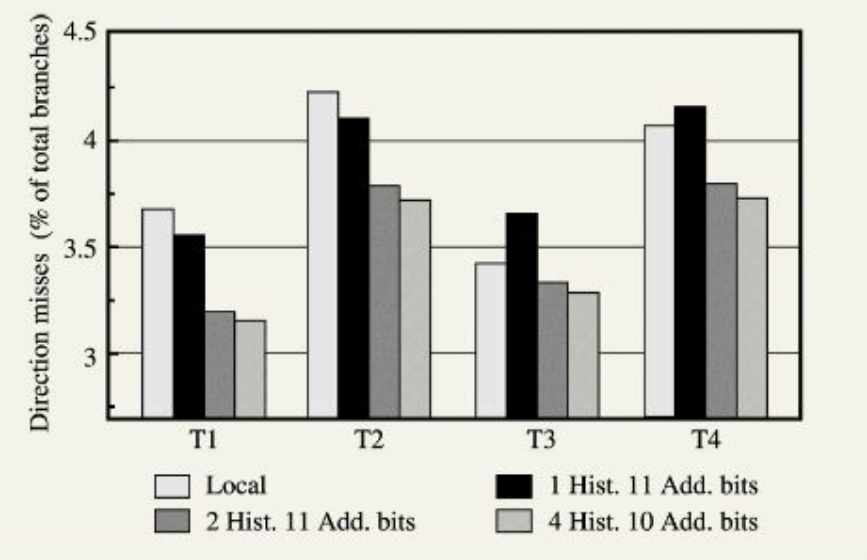


Figura 8. Evoluț ratei de predicție pentru predicția adaptivă cu 2 niveluri

Predicția direcției salturilor bazată pe istoria globală

Predicția bazată pe istoria globală a salturilor utilizează un singur registru în care se memorează istoria pentru toate salturile, în loc să se înregistreze această istorie separat pentru fiecare salt.

Pentru fiecare salt executat direcția acestuia este înregistrată în acest registru global (GR – Global Register) şi formează un tipar (pattern) global. Pentru a prezice un anumit salt trebuie luată în considerare calea prin program urmată pentru a se executa saltul.

În mod similar metodei de predicție adaptive cu 2 niveluri acest tipar global este utilizat pentru a adresa o tabelă de contori cu saturare.

Metoda de predicție globală este ilustrată în figura 9.

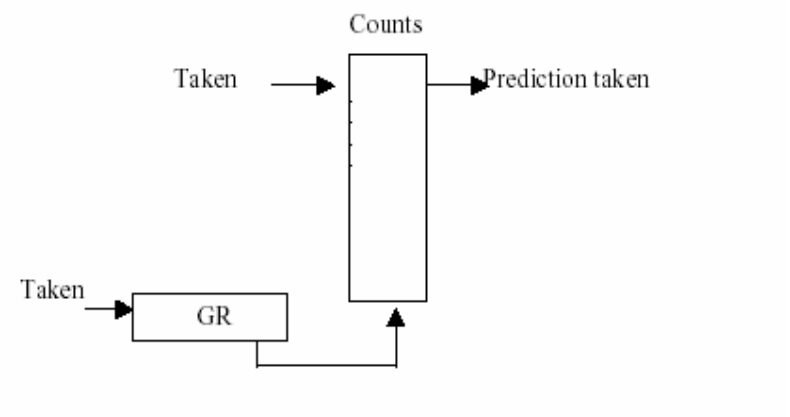


Figura 9. Predicția globală a salturilor

Avantajul metodei de predicție globale este acela că se reduce dimensiunea tabelelor utilizate pentru memorarea informațiilor necesare predicției. Se pot prezice mai multe salturi utilizînd o dimensiune de memorie specificată.

Problema care apare pentru această metodă de predicție este aceea informațiile pentru diferite salturi interferă între ele.

Există două scheme pentru predicția globală a salturilor: *gselect* şi *gshare*.

Ambele metode încearcă să resolve problema interferării informațiilor între salturi printr-o adresare mai precisă a saltului.

Metoda *gselect* este ilustrată în figura 10.

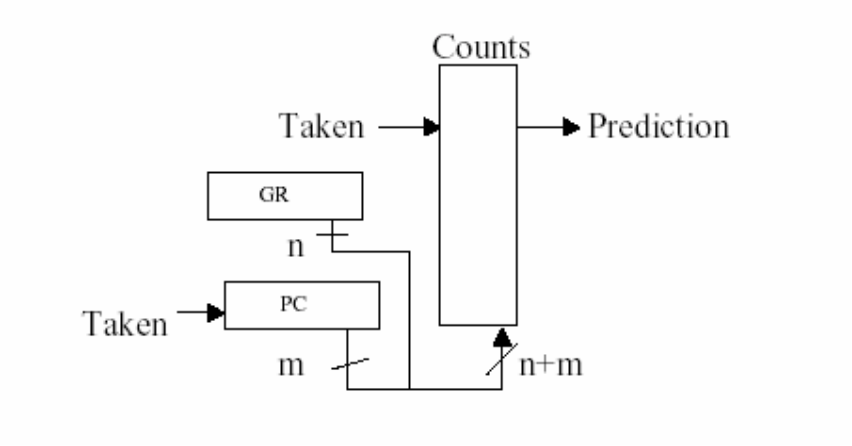


Figura 10. Metoda de predicție globală – *gselect*

Adresarea tabelei de contori se realizează utilizînd concatenarea unor biți din registrul GR şi a bitilor mai puțin semnificativi din adresa instrucțiunii de salt ca în tabelul 3.

Tabelul 3

Adresa instrucțiunii de salt	Registrul GR	Index în tabela de contori (gselect)
0000 0000	0000 0001	0000 0001
0000 0000	0000 0000	0000 0000
1111 1111	0000 0000	1111 0000
1111 1111	1000 0000	1111 0000

Metoda gshare este prezentată în figura 11.

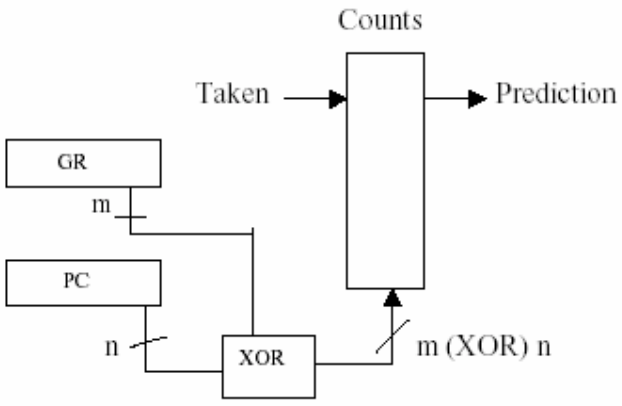


Figura 11. Metoda de predicție globală – *gshare*

Metoda *gshare* calculează indexul pentru tabela de contori ca un XOR logic între biții din registrul GR şi biții mai puțin semnificativi ai adresei instrucțiunii de salt, ca în tabelul 4.

Tabelul 4

Adresa instrucțiunii de salt	Registrul GR	Index în tabela de contori (gshare)
0000 0000	0000 0001	0000 0001
0000 0000	0000 0000	0000 0000
1111 1111	0000 0000	1111 1111
1111 1111	1000 0000	0111 1111

Metoda *gshare* elimină situațiile în care indexul pentru tabela de contori ia aceeaşi valoare pentru salturi diferite (tabelul 5).

Tabelul 5

Adresa instrucțiunii de salt	Registrul GR	Index în tabela de contori (gselect)	Index în tabela de contori (gshare)
0000 0000	0000 0001	0000 0001	0000 0001
0000 0000	0000 0000	0000 0000	0000 0000
1111 1111	0000 0000	1111 0000	1111 1111
1111 1111	1000 0000	1111 0000	0111 1111

Performanțele metodelor de predicție globală *gselect* şi *gshare* sînt illustrate în figurile 12. şi 13.

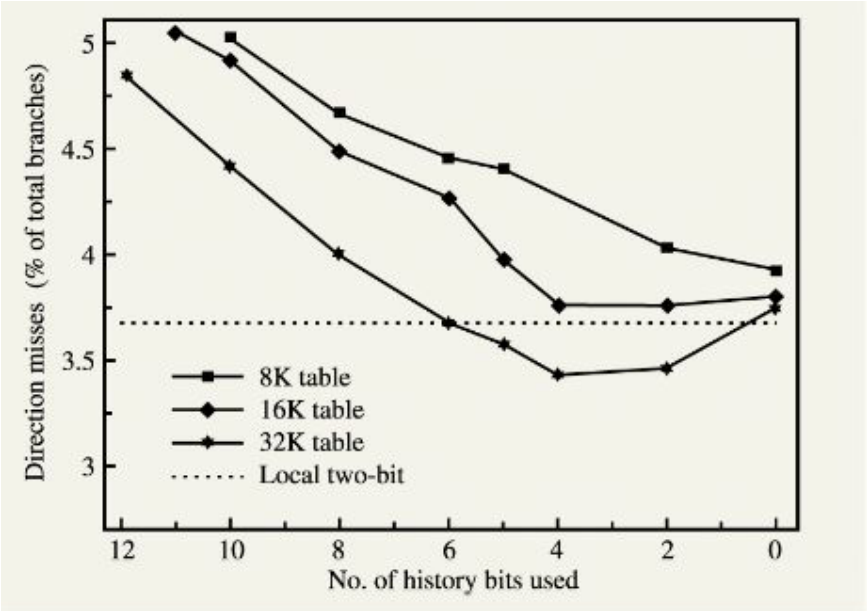
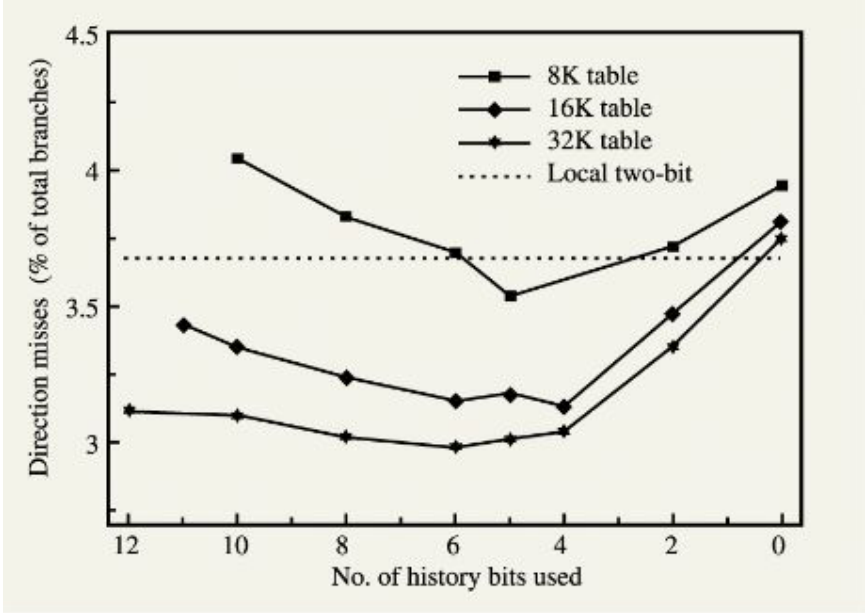


Figura 12. Rata de predicție pentru metoda de predicție globală *gselect*



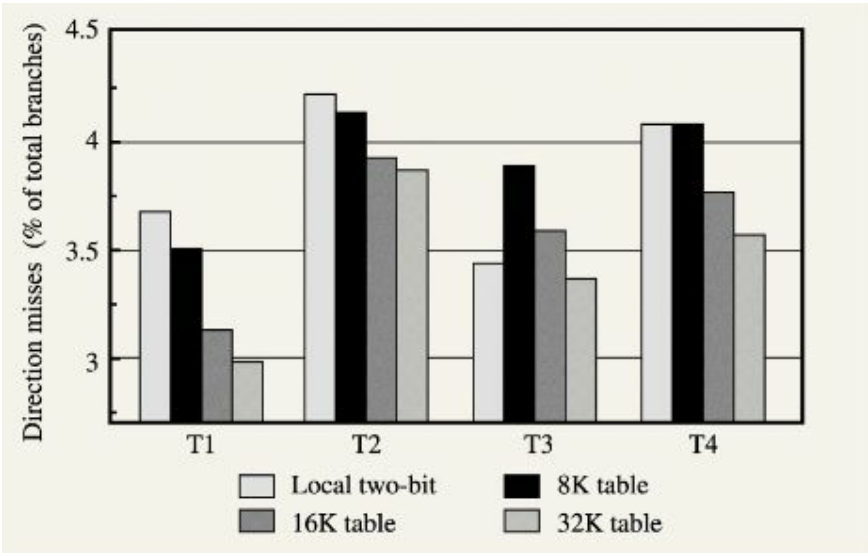


Figura 12. Rata de predicție pentru metoda de predicție globală *gshare*

În practică se utilizează metode hibride care utilizează 2 predictori (de tipuri diferite). În figura 13 este ilustrată o metodă de predicție cu selecția predictorului.

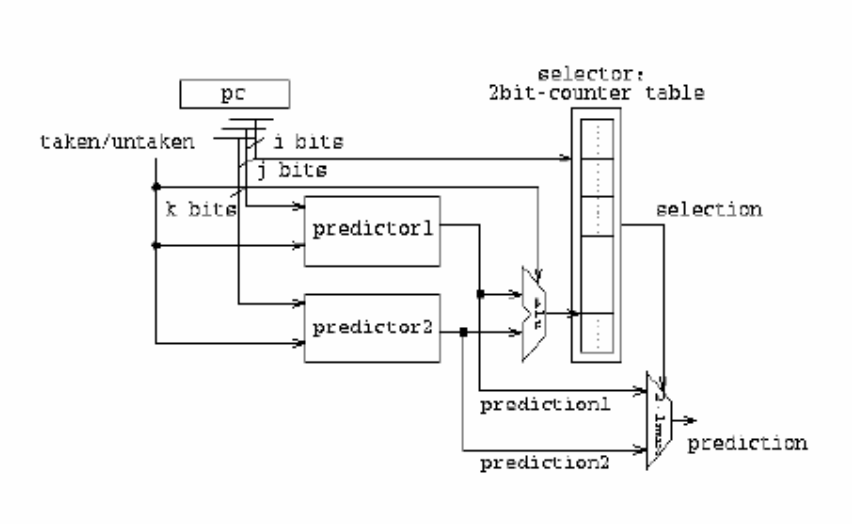


Figura 13. Metoda de predicție a salturilor cu selecția predictorului

Analiza comparativă a diferitelor metode de predicție este ilustrată în figurile 14 -

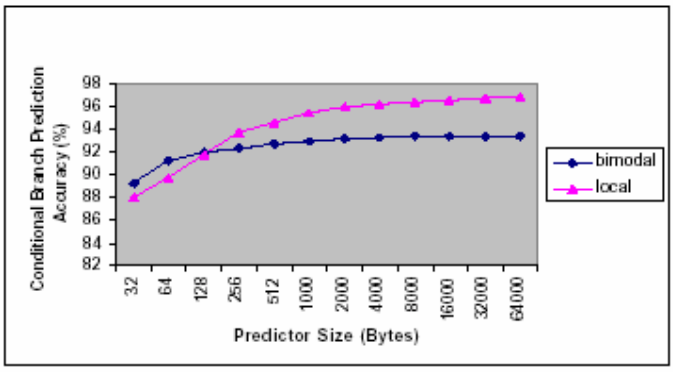


Figura 14. Comparație între predicția locală cu BTB şi predicția adaptivă cu 2 niveluri (bimodală)

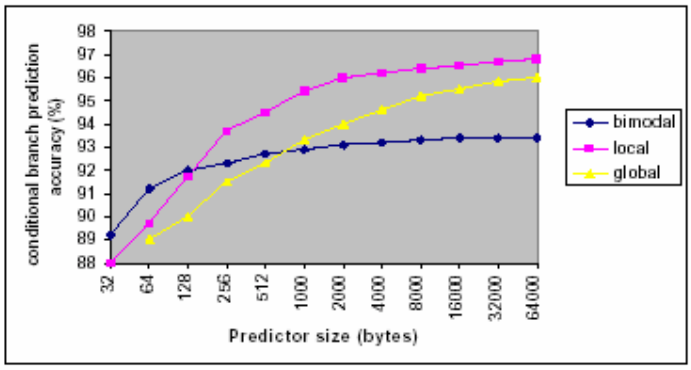


Figura 15. Comparație între predicția locală cu BTB,predicția adaptivă cu 2 niveluri (bimodală) şi predicția globală

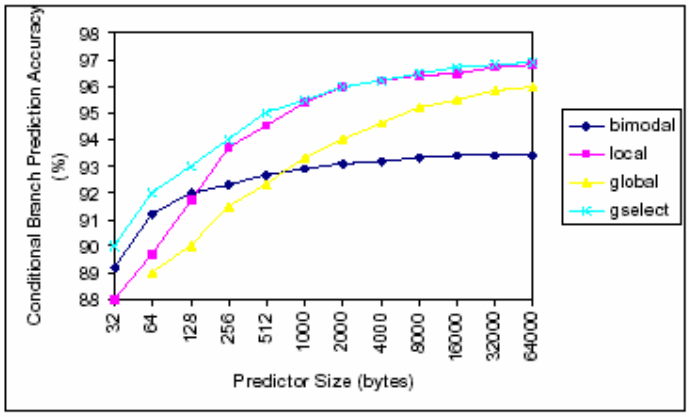


Figura 16. Comparație între predicția locală cu BTB, predicția adaptivă cu 2 niveluri (bimodală), predicția globală şi metoda *gselect*

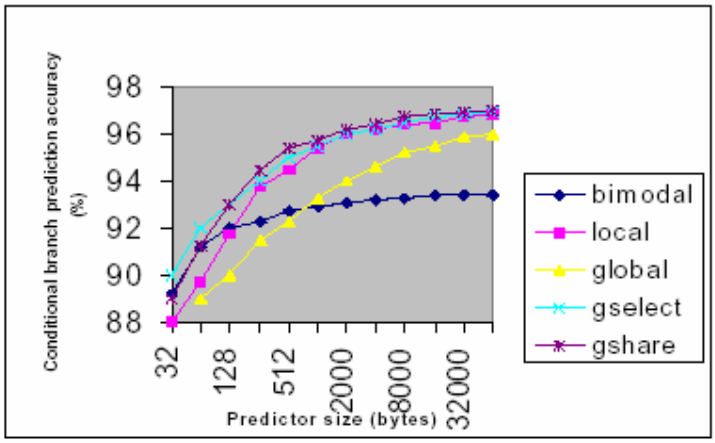


Figura 17. Comparație între predicția locală cu BTB, predicția adaptivă cu 2 niveluri (bimodală), predicția globală, metoda *gselect* şi metoda *gshare*

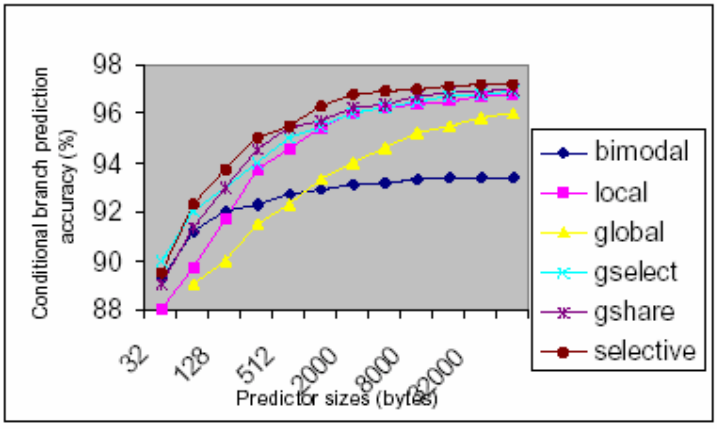


Figura 18. Comparație între predicția locală cu BTB, predicția adaptivă cu 2 niveluri (bimodală), predicția globală, metoda *gselect,* metoda *gshare* şi predicția selectivă

În concluzie predicția salturilor este influențată de următorii factori:

- identificarea corectă a saltului curent (pentru metodele de predicție globale)
- metodele de predicție trebuie să fie testate pe o gamă largă de programe
- diferite metode de predicție operează mai efficient pe diferite structuri de instrucțiuni de salt
- predicția este influențată de combinarea mai multor tipuri de predictori