ΕΛΛΑΔΑ
ΚΟΙΝΟΤΙΚΟ ΠΛΑΙΣΙΟ ΣΤΗΡΙΞΗΣ
2008
Ανάπτυξη παντού. Ανάπτυξη για όλους.

ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ ΕΠΕΑΕΚ

ΕΥΡΩΠΑΪΚΗ ΕΝΩΣΗ
ΣΥΓΧΡΗΜΑΤΟΔΟΤΗΣΗ
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Η ΠΑΙΔΕΙΑ ΣΤΗΝ ΚΟΡΥΦΗ
Επιχειρησιακό Πρόγραμμα
Εκπαίδευσης και Αρχικής
Επαγγελματικής Κατάρτισης

Σύνθεση σε Λογικό Επίπεδο

Λογική Σύνθεση

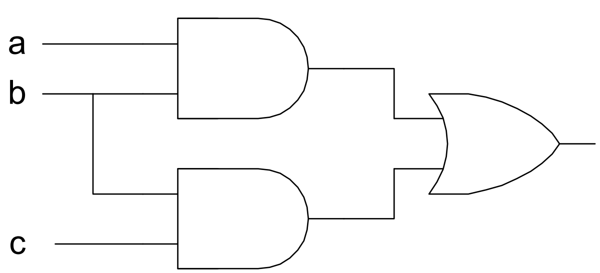
Στόχος της Λογικής Σύνθεσης και Βελτιστοποίησης για συνδυαστικά και ακολουθιακά κυκλώματα είναι ο καθορισμός της μικροσκοπικής δομής του κυκλώματος (gate-level).

Βελτιστοποίηση συνδυαστικών κυκλωμάτων

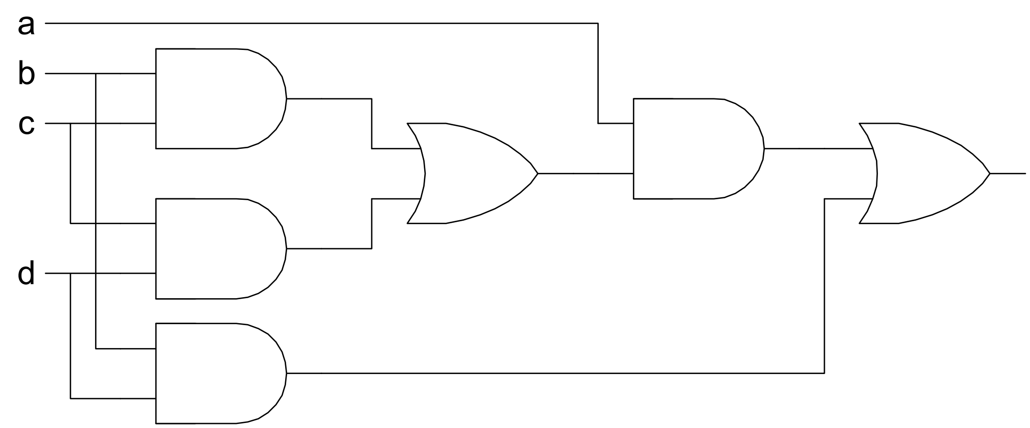
Εκφράσεις δύο επιπέδων (AND-OR) / (OR-AND)

Εκφράσεις πολλαπλών επιπέδων

F=ab+bc



F=a(bc+cd)+bd



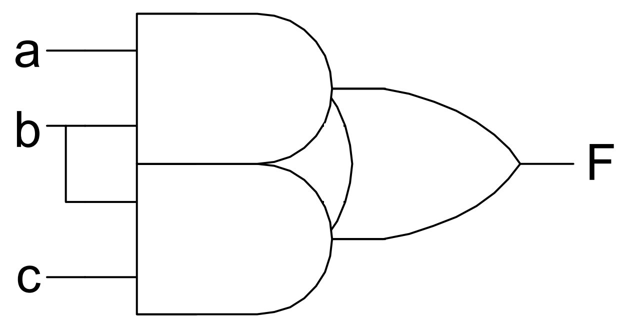
Λογική Σύνθεση

Βελτιστοποίηση εκφράσεων δύο επιπέδων

- ✓Θεωρούμε ότι τα κυκλώματα αναπαρίστανται σε μορφή αθροίσματος παραγόντων.
- ✓Οι εκφράσεις δύο επιπέδων είναι ο τυποποιημένος τρόπος αναπαράστασης λογικών συναρτήσεων
- ✓Στόχος της βελτιστοποίησης λογικών κυκλωμάτων δύο επιπέδων είναι η μείωση του μεγέθους μίας Boolean συνάρτησης αθροίσματος παραγόντων.
- ✓Μείωση μεγέθους σημαίνει: α) **μείωση των παραγόντων** και β) **μείωση των μεταβλητών**
- ✓Ανάλογα με το στυλ υλοποίησης ενός κυκλώματος οι επιμέρους στόχοι της λογικής βελτιστοποίησης διαφέρουν

Βασικές Αρχές Βελτιστοποίησης

Παράδειγμα Βελτιστοποίησης: χρήση περίπλοκων (complex) πυλών των οποίων το μέγεθος είναι ανάλογο με τον αριθμό των μεταβλητών.



Έχει καλύτερα αποτελέσματα εάν εκμεταλλευτεί και την διαμοίραση κοινών παραγόντων.

Βασικές Αρχές Βελτιστοποίησης

Ορισμοί

- Θεωρούμε συναρτήσεις με αδιάφορους όρους.
- Ορίζουμε για κάθε έξοδο τα σύνολα εισόδων:

  on-set (η έξοδος είναι 1),

  off-set (η έξοδος είναι 0) και

  dc-set (η έξοδος είναι X).

Βασικές έννοιες:

- Ελαχιστόρος (Minterm)
- Συνεπαγωγός (Implicant)
- Πρώτος Συνεπαγωγός (Prime Implicant)
- Ουσιώδης Πρώτος Συνεπαγωγός

Βασικές Αρχές Βελτιστοποίησης

Minterms

ab \ cd	00	01	11	10
00	1	1	0	1
01	1	1	0	1
11	0	0	0	1
10	0	0	0	1

0101

Implicants

ab \ cd	00	01	11	10
00	1	1	0	1
01	1	1	0	1
11	0	0	0	1
10	0	0	0	1

0*00 0*10

Prime Implicants

ab \ cd	00	01	11	10
00	1	1	0	1
01	1	1	0	1
11	0	0	0	1
10	0	0	0	1

0*0* **10

Βασικές Αρχές Βελτιστοποίησης

➢ *Αναπαριστούμε τις συναρτήσεις σαν λίστα από implicants*:

Κάθε implicant αποτελείται από τις τιμές εισόδων – εξόδων.

Κάθε είσοδος παίρνει τις τιμές 0, 1, X(*)

<u>Παράδειγμα</u>

F(a,b,c)=b΄c+ac

*01 1*1

Η ένωση των implicants αποτελεί μία κάλυψη της συνάρτησης

Βασικές Αρχές Βελτιστοποίησης

✓ *Implicant πολλαπλής εξόδου μίας Boolean συνάρτησης*:

Είναι ένα ζεύγος διανυσμάτων με ονόματα Τμήμα εισόδου/Τμήμα εξόδου.

Το τμήμα εισόδου παίρνει τιμές από το σύνολο {0, 1, *} και αναπαριστά έναν παράγοντα μεταβλητών.

Το τμήμα εξόδου παίρνει τιμές από το σύνολο {0, 1} και για κάθε έξοδο η τιμή 1 σημαίνει TRUE ή X.

Παράδειγμα

$F_1(a, b, c) = b'c + ac$ $F_2(a, b, c) = ac + ab$

(*01 10) ⇒ O Implicant ανήκει στην πρώτη συνάρτηση

(1*1 11) ⇒ O Implicant ανήκει και στις δύο

(11* 01) ⇒ O Implicant ανήκει στην δεύτερη συνάρτηση

Ορισμοί

Ελαχιστόρος πολλαπλής εξόδου: είναι ένας implicant πολλαπλής εξόδου που όλες οι είσοδες παίρνουν τιμές 0, 1 (δεν υπάρχει X).

✓ Ένας implicant πολλαπλής εξόδου αντιστοιχεί σε ένα σύνολο ελαχιστόρων της συνάρτησης και ένα υποσύνολο του dc-set.

ab \ cd	00	01	11	10
00	1	1	0	1
01	1	X	0	1
11	0	0	0	1
10	0	0	0	1

0*0*

Ελαχιστόροι: 0000, 0001, 0100

DC-set: 0101

Ορισμοί

- ✓ Με την συσχέτιση implicants και συνόλων μπορούμε να ορίσουμε έννοιες κάλυψης, τομής και δύναμης συνόλων.

<u>Παράδειγμα</u>

$f_1=a'b'c'+a'b'c+ab'c+abc+abc'=\Sigma(000, 001, 101, 111, 110)$

$f_2=a'b'c+ab'c=\Sigma(001, 101)$

Implicant β: (*01 11) → cube b΄c (001 10, 101 10, 001 01, 101 01).

Implicant γ: (1*1 10) → cube ac (101 10, 111 10).

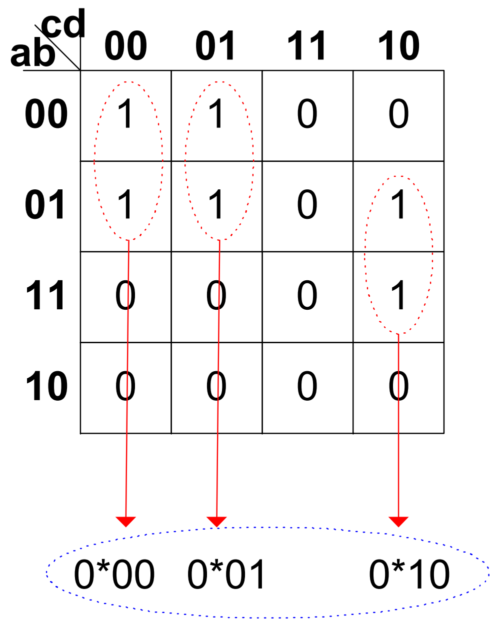
Κάλυψη: ο β καλύπτει τον ελαχιστόρο 101 10 αλλά όχι τον 111 10

Τομή: η τομή των β και γ είναι ο ελαχιστόρος 101 10.

Ορισμοί

Κάλυψη Boolean συνάρτησης:είναι ένα σύνολο F από implicants που καλύπτουν τους ελαχιστόρους της συνάρτησης f.

Μέγεθος κάλυψης συνάρτησης: είναι ο αριθμός των implicants της συνάρτησης. Οι καταστάσεις αδιαφορίας μπορούν να χρησιμοποιηθούν για να μειώσουν το μέγεθος αυτό.



Κάλυψη Συνάρτησης

Ορισμοί

Ελάχιστη κάλυψη: είναι μία κάλυψη με ελάχιστο μέγεθος.

Κόστος κάλυψης: το μέγεθός της.

ab \ cd	00	01	11	10
00	1	1	0	0
01	1	1	0	1
11	0	0	0	1
10	0	0	0	0

0*0*   *110

Ελάχιστη Κάλυψη Συνάρτησης

Παρατήρηση: στην πράξη η ελάχιστη κάλυψη δεν είναι απαραίτητα η κάλυψη ελάχιστου κόστους αφού το κόστος καθορίζεται από την υλοποίηση.

Ορισμοί

Ακριβής βελτιστοποίηση δύο επιπέδων ⇒ καθορισμός της ελάχιστης κάλυψης μίας συνάρτησης.

Ευριστική βελτιστοποίηση δύο επιπέδων ⇒ καθορισμός τοπικά ελαχίστων καλύψεων (minimal cover) αφού απαιτεί μικρότερο υπολογιστικό χρόνο και απαιτήσεις μνήμης

Ορισμοί

Τοπικά Ελάχιστη-μη πλεονάζουσα κάλυψη συνάρτησης: είναι μία κάλυψη η οποία δεν είναι υπερσύνολο οποιασδήποτε άλλης κάλυψης της συνάρτησης (η απομάκρυνση έστω ενός implicant δεν καλύπτει την συνάρτηση).

ab \ cd	00	01	11	10
00	1	1	0	1
01	1	1	0	1
11	0	0	0	0
10	0	0	0	0

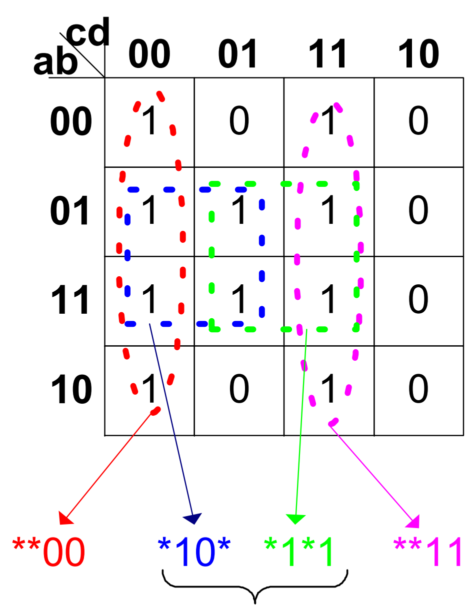
0*0* 0*10

Τοπικά Ελάχιστη Κάλυψη Συνάρτησης

Δεν είναι πλεονάζουσα αλλά ούτε και ελάχιστη αφού ο 0*10 μπορεί να μειωθεί παραπέρα.

Ορισμοί

Παράδειγμα πλεονάζουσας κάλυψης συνάρτησης



Πλεονάζουσα Κάλυψη Συνάρτησης

Ο ένας μπορεί να παραλειφθεί

Ορισμοί

Ελάχιστη κάλυψη συνάρτησης σε σχέση με την περιεκτικότητα ενός implicant: η κάλυψη στην οποία κανένας implicant δεν περιέχεται σε κάποιον άλλον implicant.

ab \ cd	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	0	0	0	0
10	0	0	0	0

Μη ελάχιστη ως προς την περιεκτικότητα

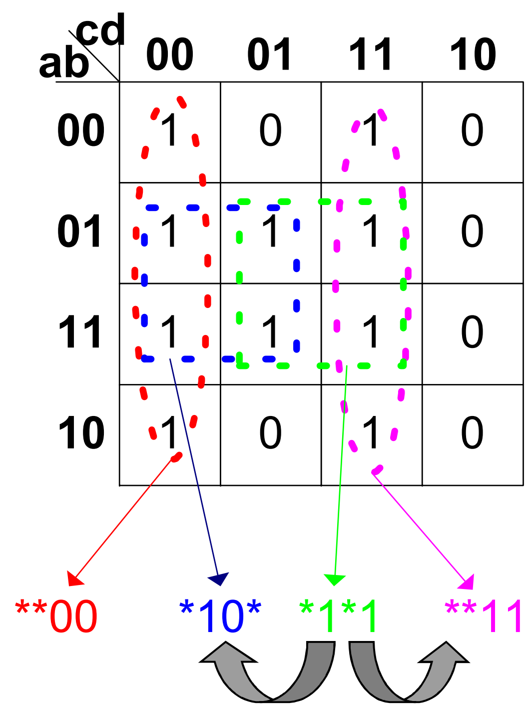
0*0*   0*01

⇓

Περιέχεται στον 0*0*

Ορισμοί

Όταν μία κάλυψη είναι ελάχιστη τότε είναι και ελάχιστη περιεκτικότητας ενός implicant. Το αντίστροφο δεν ισχύει γιατί ένας implicant μπορεί να μην περιέχεται σε άλλον implicant αλλά σε συνδυασμό άλλων implicants.



ο *1*1 περιέχεται στους *10* και **11 αφού καλύπτουν όλους τους άσσους του *1*1

Ορισμοί

<u>Παράδειγμα</u>: $f_1=a'b'c'+a'b'c+ab'c+abc+abc'$, $f_2=a'b'c+ab'c$.

Μία ελάχιστη κάλυψη είναι η ακόλουθη

$$\left.\begin{array}{l} 00*\ \ 10 \\ *01\ \ 11 \\ 11*\ \ 10 \end{array}\right\} \qquad \begin{array}{l} f_1=a'b'+b'c+ab \\ f_2=b'c \end{array}$$

Μία τοπικά ελάχιστη κάλυψη είναι η ακόλουθη

$$\left.\begin{array}{l} 00*\ \ 10 \\ *01\ \ 01 \\ 1*1\ \ 10 \\ 11*\ \ 10 \end{array}\right\} \qquad \begin{array}{l} \text{Τοπικά} \\ \text{Ελάχ.} \end{array}$$

Ορισμοί

Μία ελάχιστη κάλυψη σε σχέση με την περιεκτικότητα μονού implicant είναι η ακόλουθη

00* 10
*01 01
*01 10
1*1 10
11* 10

Ελάχιστη σε σχέση με την περιεκτικότητα μονού implicant*

*. Ο τρίτος implicant περιέχεται στην ένωση του πρώτου και του τέταρτου και για αυτό είναι **πλεονάζουσα** η κάλυψη.

Ορισμοί

Πρωταρχικός (Prime) Implicant: ένας implicant είναι πρωταρχικός εάν δεν περιέχεται από κανέναν implicant της συνάρτησης.

Πρωταρχική κάλυψη: μία κάλυψη που όλοι οι implicant είναι πρωταρχικοί.

Ουσιώδης πρωταρχικός implicant. Καλύπτει έναν ελαχιστόρο της συνάρτησης που δεν καλύπτεται από άλλον implicant.

ab \ cd	00	01	11	10
00	0	1	0	0
01	0	1	1	1
11	1	1	1	0
10	0	0	1	0

*1*1

Μη ουσιώδης πρωταρχικός inplicant

Οι υπόλοιποι είναι ουσιώδης

Ακριβής ελαχιστοποίηση

Η ακριβής ελαχιστοποίηση (exact minimization) ασχολείται με τον υπολογισμό της ελάχιστης κάλυψης.

Θεώρημα. Υπάρχει ελάχιστη κάλυψη η οποία είναι πρωταρχική.

Με χρήση του θεωρήματος περιορίζεται η αναζήτηση μόνο σε καλύψεις που αποτελούνται από ουσιώδεις implicants (Μέθοδος Quine-McCluskey).

Πίνακας Prime Implicants: είναι ένας πίνακας A δυαδικών τιμών με: (α) τις στήλες να αντιστοιχούν στους prime implicants,

(β) γραμμές να αντιστοιχούν στους ελαχιστόρους της συνάρτησης.

(γ) Όταν $a_{ij}=1$ τότε ο j prime implicant καλύπτει τον ελαχιστόρο i.

Ελάχιστη κάλυψη x= ελάχιστο σύνολο στηλών (prime implicants) που καλύπτει όλες τις γραμμές (ελαχιστόρους).

Συνθήκη $Ax \geq 1$, x διάνυσμα 1xn

Ακριβής ελαχιστοποίηση

Παράδειγμα

$F=a'b'c'+a'b'c+ab'c+abc+abc' = \Sigma(000, 001, 101, 111, 110)$

με prime implicants τους α(00* 1), β(*01 1), γ(1*1 1), δ(11* 1). Τότε ο πίνακας A των prime implicants είναι:

	α	**β**	**γ**	**δ**
000	1	0	0	0
001	1	1	0	0
101	0	1	1	0
111	0	0	1	1
110	0	0	0	1

Η ζητούμενη κάλυψη είναι η {α, β, δ}, δηλ. $x=[1101]^T$. Ισχύει $Ax\geq 1$

Εναλλακτική κάλυψη είναι και η {α, γ, δ}, δηλ. $x=[1011]^T$.

Ακριβής ελαχιστοποίηση

Λύση προβλήματος: εύρεση πίνακα και υπολογισμός ελάχιστης κάλυψης.

Δυσκολίες: αδυναμία προσέγγισης προβλήματος κάλυψης (εκθετική πολυπλοκότητα) και εκθετικό μέγεθος πίνακα των prime implicants.

Βελτίωση με μείωση του πίνακα:

- ***Αφαίρεση ουσιωδών στηλών και καλυπτόμενων γραμμών***: μία ουσιώδης στήλη αντιστοιχεί σε έναν ουσιώδη πρωταρχικό implicant που πρέπει να είναι μέρος κάθε λύσης.

	α	**β**	**γ**	**δ**
000	1	0	0	0
001	1	1	0	0
101	0	1	1	0
111	0	0	1	1
110	0	0	0	1

Οι α, δ επιλέγονται ως ουσιώδεις αφού καλύπτουν τους 000 και 110 που δεν καλύπτονται από άλλους

	β	**γ**
101	1	1

Ακριβής ελαχιστοποίηση

- ***Αφαίρεση κυρίαρχων γραμμών***: Όταν μία γραμμή (ελαχιστόρος) κυριαρχεί μίας άλλης, τότε και οι δύο έχουν κοινούς άσσους με την κυρίαρχη να έχει μερικούς επιπλέον.

Εάν καλυφθεί η κυριαρχούμενη τότε σίγουρα καλύπτεται και η κυρίαρχη

Η κυρίαρχη μπορεί να απαλοιφθεί

	α	**β**	**γ**	**δ**
000	1	1	0	0
~~001~~	~~1~~	~~1~~	~~1~~	~~0~~
101	0	1	1	0
111	0	0	1	1
110	0	0	0	1

Η γραμμή 001 είναι κυρίαρχη της 000 και μπορεί να απαλειφθεί αφού για την 000 θα επιλεγεί είτε ο implicant α είτε ο β που καλύπτει και την 001

Ακριβής ελαχιστοποίηση

- *Αφαίρεση κυριαρχούμενων στηλών*: μία στήλη που είναι κυρίαρχη μίας άλλης έχει άσσους τουλάχιστον στους ελαχιστόρους της κυριαρχούμενης οπότε την καλύπτει.

	α	β	γ	δ
000	1	1	0	0
001	1	1	1	0
101	0	1	1	0
111	0	0	1	1
110	0	0	0	1

Ο implicant α κυριαρχείται από τον β και διαγράφεται καθώς ο β καλύπτει όλους τους ελαχιστόρους του α και κάποιους επιπλέον, οπότε αρκεί να επιλεγεί ο β.

Ακριβής ελαχιστοποίηση

- Εάν ο πίνακας είναι κενός μετά τις ελαχιστοποιήσεις τότε οι ουσιώδεις implicants καλύπτουν την συνάρτηση.
- Σε αντίθετη περίπτωση επιλέγονται διαφορετικοί συνδυασμοί και προσδιορίζεται το κόστος τους.

$\Downarrow$

Μέθοδος Petrick

- Σχηματισμός αθροισμάτων implicants (ένα για κάθε γραμμή).
- Δημιουργία έκφρασης με το γινόμενο των αθροισμάτων.
- Μετατροπή σε αθροίσματα γινομένων με άλγεβρα Boole (εκθετικός αριθμός λειτουργιών). Επιλογή παράγοντα με το μικρότερο κόστος.

Πχ. $\alpha(\alpha+\beta)(\beta+\gamma)(\gamma+\delta)\delta=1 \Rightarrow \alpha\beta\delta+\alpha\gamma\delta=1$ (2 ελάχιστες καλύψεις μεγέθους 3)

Ακριβής ελαχιστοποίηση

Παράδειγμα μεθόδου Petrick

(χωρίς ελαχιστοποιήσεις πίνακα)

	α	**β**	**γ**	**δ**
000	1	0	0	0
001	1	1	0	0
101	0	1	1	0
111	0	0	1	1
110	0	0	0	1

$$\alpha(\alpha+\beta)(\beta+\gamma)(\gamma+\delta)\delta=1 \Rightarrow \alpha\beta\delta+\alpha\gamma\delta=1$$

(2 ελάχιστες καλύψεις μεγέθους 3)

Ακριβής ελαχιστοποίηση

Μέθοδος Espresso-Exact

- Οι κυριότερες βελτιώσεις σε σχέση με την μέθοδο Quine-McCluskey είναι δύο:

  α) κατασκευή μικρότερου μειωμένου πίνακα ουσιωδών implicants και

  β) χρήση ενός αποδοτικού branch and bound αλγορίθμου για την κάλυψη.

- Διαμέριση των πρωταρχικών implicants σε τρία σύνολα:

  (α) **Ουσιώδεις**: οι γνωστοί,

  (β) **Ολικώς πλεονάζοντες** είναι όσοι καλύπτονται από τους ουσιώδεις και το σύνολο dc.

  (γ) Οι υπόλοιποι είναι οι **μερικώς πλεονάζοντες** και αντιστοιχούν στις στήλες του μειωμένου πίνακα implicants.

Ακριβής ελαχιστοποίηση

Παράδειγμα:

Έστω μία συνάρτηση με τους ακόλουθους ελαχιστόρους:(0000 1), (0010 1), (0100 1), (0110 1), (1000 1), (1010 1), (0101 1), (0111 1), (1001 1), (1011 1), (1101 1). Οι πρωταρχικοί implicants είναι οι:

α 0**0 ⟶ 0000, 0010, 0100, 0110
β *0*0 ⟶ 0000, 0010, 1000, 1010
γ 01** ⟶ 0100, 0101, 0110, **0111**
δ 10** ⟶ 1000, 1001, 1010, **1011**
ε 1*01 ⟶ 1001, 1101
ζ *101 ⟶ 0101, 1101

Οι πρωταρχικοί implicants γ, δ είναι ουσιώδεις επειδή καλύπτουν τους ελαχιστόρους 0111 1 και 1011 1. Οι υπόλοιπο είναι μερικώς πλεονάζοντες.

Ο μειωμένος πίνακας είναι:

	α	β	ε	ζ
0000, 0010	1	1	0	0
1101	0	0	1	1

Λύση: (α ή β) και (ε ή ζ) και (γ και δ) = αεγδ, αζγδ, βεγδ, βζγδ

Ευριστική ελαχιστοποίηση

Συνήθως χρησιμοποιείται για τις πρακτικές εφαρμογές.

- Τα παλαιότερα προγράμματα υπολόγιζαν όλους τους πρωταρχικούς implicants και χρησιμοποιούσαν ευριστικούς αλγορίθμους για τον υπολογισμό της κάλυψης.
- Τα πιο πρόσφατα προγράμματα δεν υπολογίζουν όλους τους πρωταρχικούς implicants για μείωση του αποθηκευτικού χώρου.
- Αντίθετα, υπολογίζουν μία πρωταρχική κάλυψη από τον προσδιορισμό της συνάρτησης.
- Κατόπιν διαχειρίζονται την κάλυψη μετατρέποντας και/ή διαγράφοντας implicants έως ότου μία ελάχιστη (τοπικά) κάλυψη βρεθεί (Στρατηγική επαναληπτικής βελτίωσης).

Ευριστική ελαχιστοποίηση

✓ Η ευριστική κάλυψη είναι ουσιαστικά η εφαρμογή ενός συνόλου τελεστών στην λογική κάλυψη.

✓ Όταν οι τελεστές δεν μπορούν να μειώσουν την κάλυψη παραπέρα, έχει βρεθεί μία τοπικά ελάχιστη κάλυψη:

Τελεστής Expand.

➢ Κάνει μία κάλυψη πρωταρχική και ελάχιστη σε σχέση με την περιεκτικότητα ενός implicant.

➢ Διαχειρίζεται τους implicants έναν προς έναν και τους μετατρέπει σε πρωταρχικούς (τους αντικαθιστά με άλλους που τους περιέχουν).

➢ Όσοι άλλοι καλύπτονται, διαγράφονται.

Ευριστική ελαχιστοποίηση

Τελεστής Reduce.

- Μετατρέπει μία κάλυψη σε μη πρωταρχική με το ίδιο μέγεθος.
- Προσπαθεί να αντικαταστήσει κάθε implicant με έναν από αυτούς που περιέχει με την συνθήκη να συνεχίσει να καλύπτεται η συνάρτηση.

Τελεστής Reshape.

- Μετατρέπει την κάλυψη ενώ διατηρεί το μέγεθος.
- Διαχειρίζεται τους implicants σε ζεύγη όπου ο ένας γίνεται expanded και ο άλλος reduced με συνθήκη ότι συνεχίζει να καλύπτεται η συνάρτηση.

Τελεστής Irredudant.

- Κάνει μία κάλυψη μη πλεονάζουσα.
- Ένα ελάχιστο υποσύνολο από implicants επιλέγεται όπου κανένας implicant στο υποσύνολο δεν καλύπτεται από τους υπόλοιπους.

Ευριστική ελαχιστοποίηση

Ο χαρακτηρισμός των ευριστικών προγραμμάτων μπορεί να γίνει με βάση τους τελεστές που χρησιμοποιούν και την σειρά τους:

Πρόγραμμα PRESTO:

- Εφαρμογή μία φορά του τελεστή Expand.
- Πετυχαίνεται πρωταρχικότητα και ελαχιστοποίηση με βάση την περιεκτικότητα μονού implicant.
- Μη ικανοποιητική κάλυψη.

Πρόγραμμα MINI:

- Επαναλαμβάνει την εφαρμογή των τελεστών expand, reduce και reshape.
- Η πρωταρχική (prime) κάλυψη επιτυγχάνεται με τον expand και μεταβάλλεται από τους άλλους δύο τελεστές για την επίτευξη μικρότερης λύσης.

Ευριστική ελαχιστοποίηση

- Οι επαναλήψεις σταματούν όταν δεν βελτιώνεται η λύση.

Πρόγραμμα ESPRESSO:

- Εφαρμογή του τελεστή *Expand* για την δημιουργία πρωταρχικής κάλυψης.
- Εφαρμογή του τελεστή *Irredundant* που εξασφαλίζει κάλυψη μη πλεονάζουσα.
- Επανάληψη τελεστών *Reduce*, *Expand* και *Irredundant* για βελτίωση κάλυψης.

- ✓ Εκτός από την σειρά εφαρμογής των τελεστών τα ευριστικά προγράμματα διαφέρουν και στον τρόπο υλοποίησής τους.
- ✓ Χρησιμοποιούν ευριστικές λύσεις (πχ η σειρά επεξεργασίας των implicants από τον τελεστή Expand επηρεάζει το μέγεθος της κάλυψης).

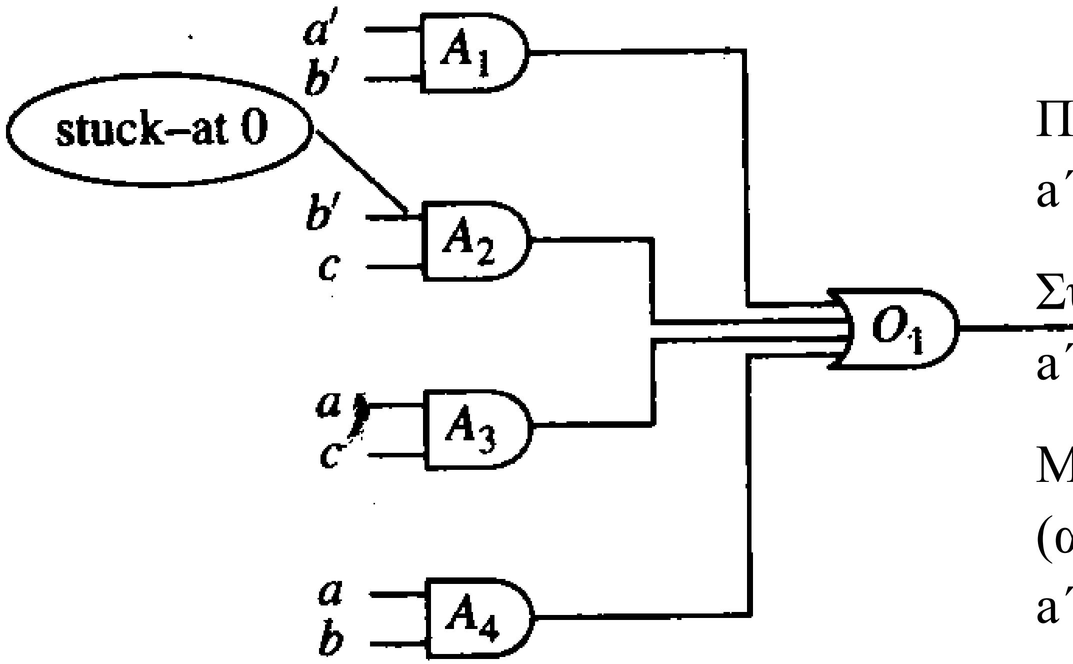
Μερικοί αλγόριθμοι έχουν μεγαλύτερη από πολυωνυμική πολυπλοκότητα.

Ιδιότητες ελεγξιμότητας (testability)

- ✓ Μετά την κατασκευή του ένα κύκλωμα πρέπει να ελεγχθεί για δυσλειτουργίες
- ✓ Οι δυσλειτουργίες αναπαρίσταται με τα μοντέλα σφαλμάτων (stuck-at 0/1).
- ✓ Η ελεγξιμότητα ενός κυκλώματος είναι η ευκολία ελέγχου του και είναι μία μετρική ποιότητας του.
- ✓ Ένα πλήρως ελέγξιμο κύκλωμα είναι αυτό του οποίου κάθε σφάλμα μπορεί να ανιχνευτεί από κάποιο διάνυσμα εισόδου.
- ✓ Η ανίχνευση ενός σφάλματος σχετίζεται με τις controlability/observability.

Θεώρημα. Απαραίτητη και ικανοποιητική συνθήκη για την πλήρη ελεγξιμότητα ενός κυκλώματος (μονών σφαλμάτων μόνιμης τιμής) είναι ότι πρέπει η κάλυψη (AND-OR) να είναι πρωταρχική και μη πλεονάζουσα.

Ιδιότητες ελεγξιμότητας (testability)



Πλεονάζουσα κάλυψη:
a΄b΄+b΄c+ac+ab

Συνάρτηση σφάλματος:
a΄b΄+ac+ab

Μη πλεονάζουσα κάλυψη
(αφαίρεση πύλης A_3):
a΄b΄+b΄c+ab

Βελτιστοποίηση συναρτήσεων πολλών επιπέδων

Βελτιστοποίηση κυκλ. πολλαπλών επιπέδων

- ✓ Πολλές φορές τα κυκλώματα υλοποιούνται με λογική πολλαπλών επιπέδων.
- ✓ Είναι πολύ ευέλικτη η υλοποίηση με πολλαπλά επίπεδα αλλά έχει το μειονέκτημα της δυσκολίας μοντελοποίησης και βελτιστοποίησης.
- ✓ Πολύ λίγες μέθοδοι **<u>ακριβούς</u>** βελτιστοποίησης έχουν βρεθεί και έχουν το μειονέκτημα της υψηλής υπολογιστικής πολυπλοκότητας.
- ✓ Οι ευριστικές μέθοδοι έχουν αποδειχτεί αποτελεσματικές στην μείωση της επιφάνειας και αύξηση της ταχύτητας μεγάλων κυκλωμάτων.
- ✓ Η γνώση που υπάρχει για τις υλοποιήσεις πολλών επιπέδων στην βελτιστοποίηση είναι πολύ μικρότερη από αυτήν των δύο επιπέδων.
- ✓ Χρησιμοποιούμε το μοντέλο του αφηρημένου δικτύου όπου έχουμε την διασύνδεση συνδυαστικών πυλών μονής εξόδου χωρίς αναδράσεις.

Βελτιστοποίηση κυκλ. πολλαπλών επιπέδων

Τα λογικά δίκτυα μπορούν να υλοποιηθούν ανάλογα με τα διάφορα σχεδιαστικά στυλ που σχετίζονται με τον τύπο των πυλών που διασυνδέονται:

✓***Υλοποίηση με μονό τύπο πυλών***: Nors, Nands με όριο fan-in/fan-out

✓***Υλοποίηση με πύλες βιβλιοθήκης***: χρησιμοποιούνται πύλες που υπάρχουν σε ειδική βιβλιοθήκη υλοποίησης.

✓***Macro-cell υλοποίηση***: δομές που υλοποιούν περίπλοκες λογικές συναρτήσεις.

Βελτιστοποίηση κυκλ. πολλαπλών επιπέδων

Η επιλογή του σχεδιαστικού στυλ επηρεάζει τις μεθόδους σύνθεσης και βελτιστοποίησης που γίνονται ευριστικά σε δύο στάδια:

1. Χρησιμοποιείται αρχικά ένα γενικό μοντέλο για την βελτιστοποίηση του λογικού δικτύου που δεν λαμβάνει υπόψη περιορισμούς υλοποίησης.

2. Binding: Λαμβάνονται υπόψη οι περιορισμοί υλοποίησης (πυλών).

Μοντέλα & Μετασχηματισμοί

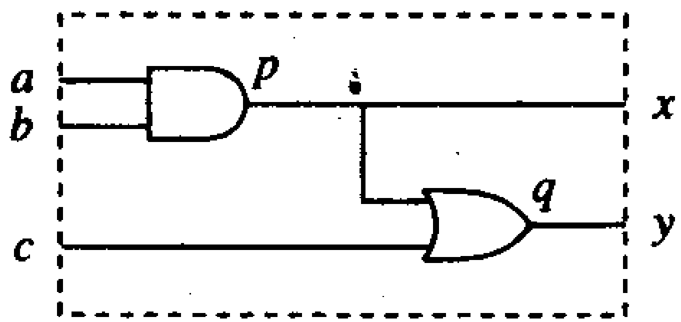
Η συμπεριφορά ενός συνδυαστικού δικτύου n-εισόδων και m-εξόδων μπορεί να εκφραστεί από έναν πίνακα μερικώς ή ολικώς προσδιορισμένων δυαδικών συναρτήσεων που αναπαριστούν την αντιστοίχιση του συνόλου εισόδων στις εξόδους.

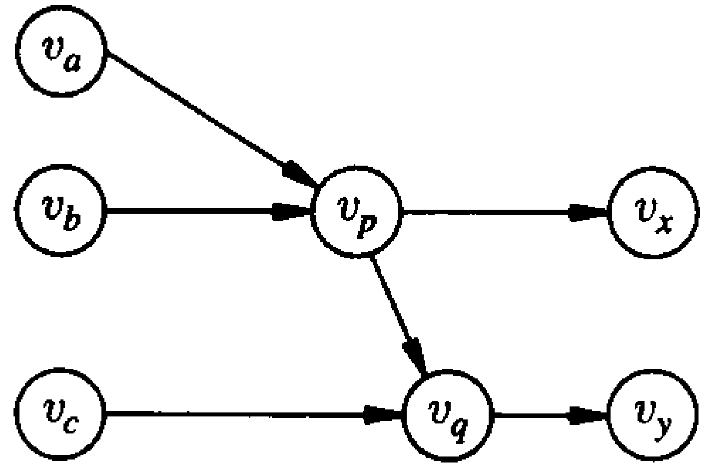
Λογικό δίκτυο:

- ✓Είναι μία δομή που συσχετίζει τα modules (I/O θύρες, πύλες) με γραμμές διασύνδεσης.
- ✓Αναπαρίσταται με έναν άκυκλο κατευθυνόμενο γράφο με τις κορυφές να αντιστοιχούν στα modules και τις ακμές στις γραμμές διασύνδεσης.
- ✓Εάν τα modules αντιστοιχούν ακριβώς σε πύλες τότε το δίκτυο ονομάζεται mapped network.
- ✓Η συμπεριφορά ενός κυκλώματος μπορεί να αντιστοιχηθεί σε πολλές ισοδύναμες δομές.

Μοντέλα & Μετασχηματισμοί

✓Πολλές φορές είναι αδύνατον να εξαχθεί η περιγραφή συμπεριφοράς από ένα δίκτυο (ακόμη και σχετικά μικρό) λόγω υπερβολικού μεγέθους.





x=ab

y=c+ab

✓Τα εσωτερικά modules περιγράφουν βαθμωτές συνδυαστικές συναρτήσεις,

✓Οι διασυνδέσεις περιγράφουν την κατασκευή.

✓Τα λογικά δίκτυα είναι πολύ χρήσιμα αφού η εσωτερική τοπική περιγραφή συναρτήσεων αν και απλή βοηθάει στην μοντελοποίηση πολύπλοκων συμπεριφορών.

Μοντέλα & Μετασχηματισμοί

- ✓ Οι βελτιστοποιήσεις εφαρμόζονται σε μη-ιεραρχικά δίκτυα (η μετατροπή ιεραρχικών δικτύων είναι τετριμμένη διαδικασία.
- ✓ Θεωρούμε ότι οι γραμμές διασύνδεσης είναι δύο σημείων (αρχή-τέλος/χωρίς fanout) ώστε να υπάρχει ισομορφία του δικτύου με τον γράφο.

Μοντέλα & Μετασχηματισμοί

Μη ιεραρχικό συνδυαστικό λογικό δίκτυο:

1. Ένα σύνολο κορυφών διαιρεμένο σε τρία υποσύνολα:

   (α) κύριες είσοδοι,

   (β) κύριες έξοδοι,

   (γ) εσωτερικές κορυφές (κάθε κορυφή έχει το όνομα μίας μεταβλητής).

2. Ένα σύνολο βαθμωτών συνδυαστικών συναρτήσεων Boole.

   Οι συναρτήσεις αντιστοιχούν σε κάθε εσωτερική κορυφή.

   Οι μεταβλητές υποστήριξης κάθε τοπικής συνάρτησης είναι μεταβλητές που σχετίζονται με κύριες εισόδους ή εσωτερικές κορυφές.

3. Ένα σύνολο αναθέσεων των κυρίων εξόδων σε εσωτερικές κορυφές που δηλώνουν ποιες μεταβλητές είναι απευθείας ορατές στις εξόδους.

Μοντέλα & Μετασχηματισμοί

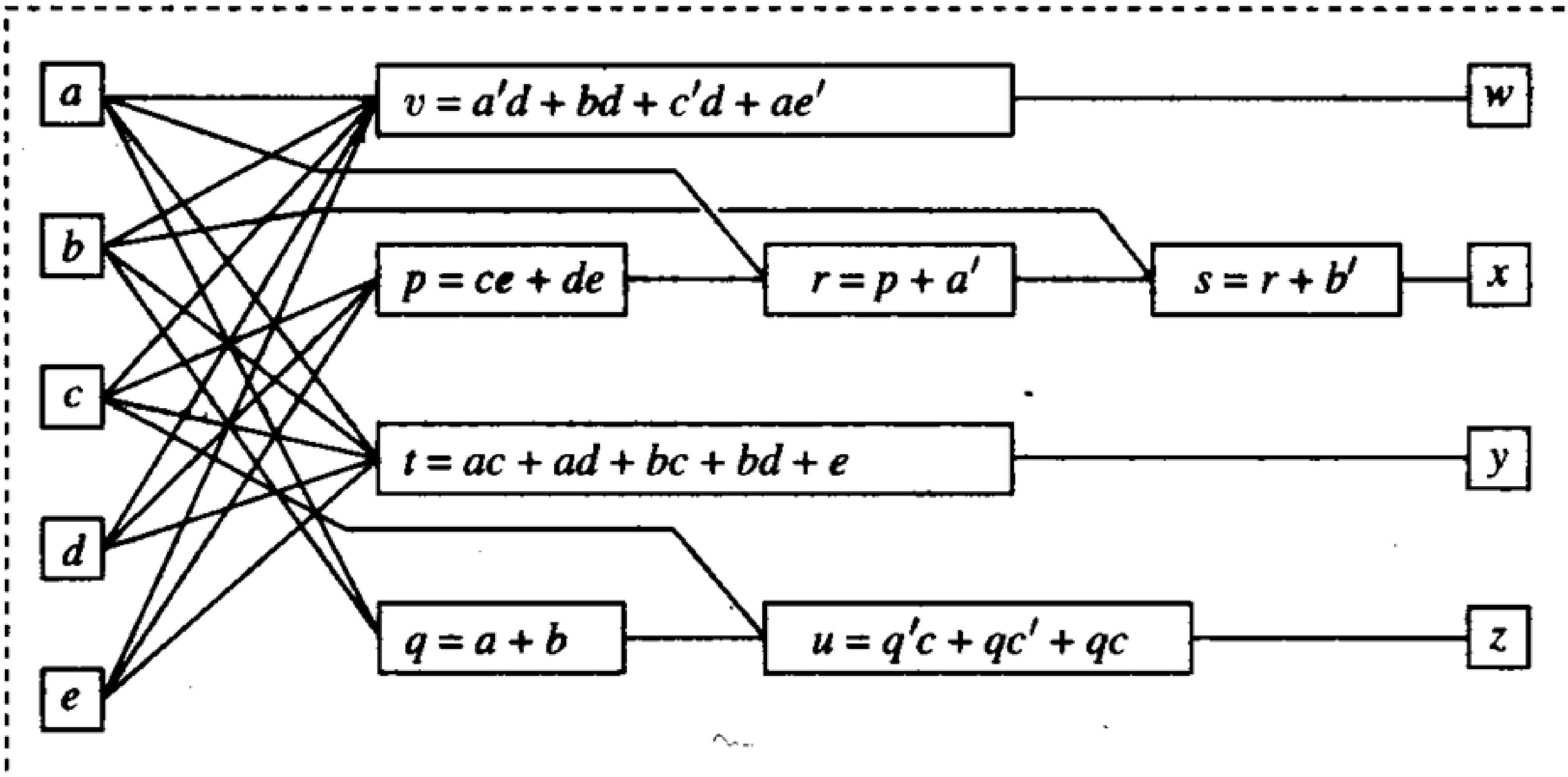
- ✓ Κάθε κορυφή και αντίστοιχα τοπική συνάρτηση σχετίζεται με μία μη-συμπληρωμένη μεταβλητή.
- ✓ Οι αντιστροφείς υπονοούνται και δεν αναπαρίστανται.
- ✓ Κάθε κορυφή παρέχει σήματα κανονικά και ανεστραμμένα και το δίκτυο ονομάζεται διπλής πολικότητας.
- ✓ Η μετατροπή του σε δίκτυο μονής πολικότητας με την χρήση αντιστροφέων μπορεί να γίνει με την προσθήκη κορυφών και ακμών.

<u>Παράδειγμα</u>:

$$f = \begin{pmatrix} a'd + bd + c'd + ae' \\ a' + b' + ce + de \\ ac + ad + bc + bd + e \\ a + b + c \end{pmatrix} = \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix}$$

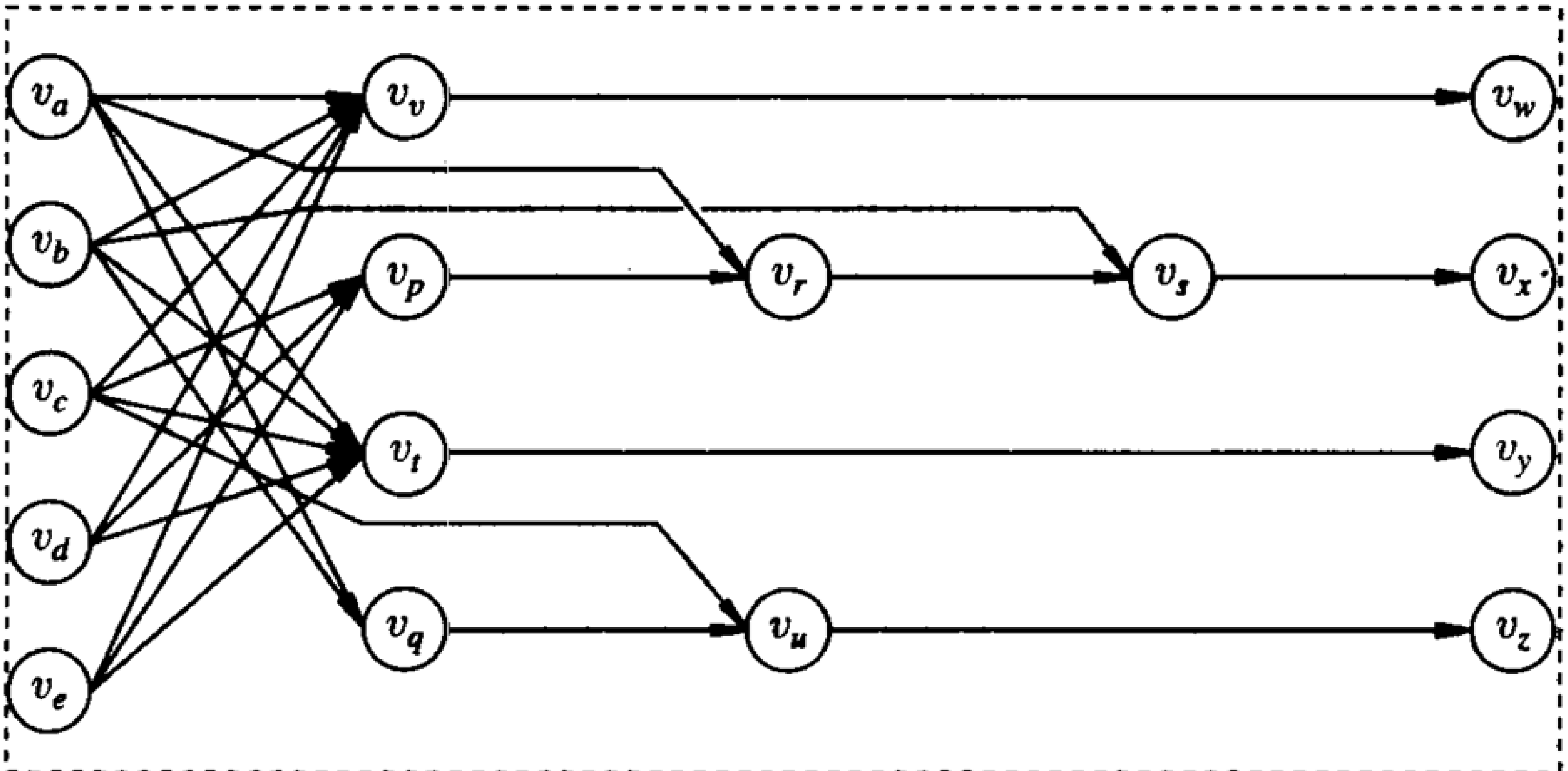
$$\left.\begin{aligned} p &= ce + de \\ q &= a + b \\ r &= p + a' \\ s &= r + b' \\ t &= ac+ad+bc+bd+e \\ u &= q'c+qc'+qc \\ v &= a'd+bd+c'd+ae' \end{aligned}\right\} \text{Εσωτερικές κορυφές}$$

Μοντέλα & Μετασχηματισμοί



(a)

Μοντέλα & Μετασχηματισμοί



(b)

Βελτιστοποίηση Λογικών Δικτύων

✓Κυριότεροι στόχοι βελτιστοποίησης:

(α) μείωση επιφάνειας,

(β) μείωση καθυστέρησης και

(γ) ελεγξιμότητα του κυκλώματος.

✓Στην διεπίπεδη υλοποίηση αθροίσματος παραγόντων η επιφάνεια και καθυστέρηση είναι ανάλογες με το μέγεθος της κάλυψης:

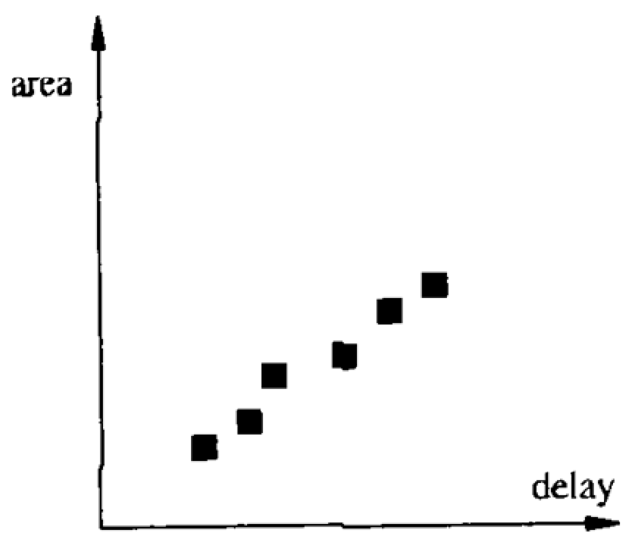
Ελαχιστοποίηση της κάλυψης αντιστοιχεί σε βελτιστοποίηση επιφάνειας και ταχύτητας.

Επίτευξη μη πλεονάζουσας κάλυψης αντιστοιχεί σε μεγιστοποίηση της ελεγξιμότητας.

Βελτιστοποίηση Λογικών Δικτύων

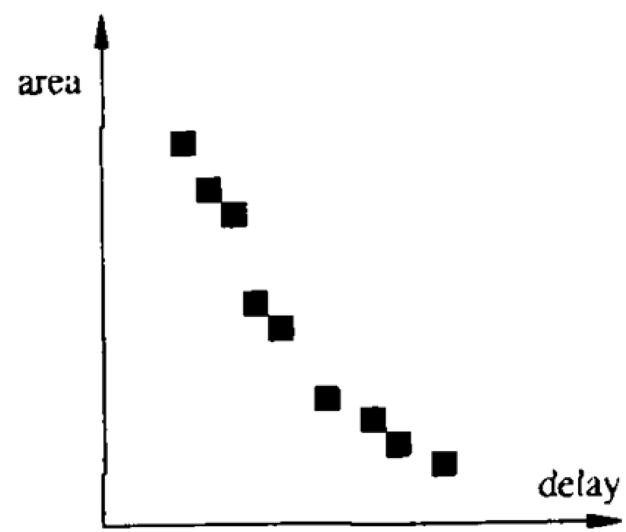
✓Στα κυκλώματα πολλαπλών επιπέδων οι υλοποιήσεις ελάχιστης επιφάνειας δεν αντιστοιχούν σε βέλτιστη ταχύτητα (Trade-Off). Πχ αθροιστές ripple carry και lookahead.

2-επίπεδη



(a)

πολυ-επίπεδη



(b)

Βελτιστοποίηση Λογικών Δικτύων

Είδη βελτιστοποίησης:

- Βελτιστοποίηση επιφάνειας (με περιορισμούς στην καθυστέρηση).
- Βελτιστοποίηση καθυστέρησης (με περιορισμούς στην επιφάνεια).

Οι περιορισμοί καθυστέρησης μπορούν να εκφραστούν με χρόνους αφίξεων εισόδων και απαιτούμενους χρόνους εξόδων.

Βελτιστοποίηση Λογικών Δικτύων

Επιφάνεια Δικτύου

- ✓ Η επιφάνεια ενός δικτύου πολλαπλών επιπέδων οφείλεται σε λογικές πύλες και διασυνδέσεις.
- ✓ Όταν το δίκτυο υλοποιεί πύλες βιβλιοθήκης τότε η επιφάνεια κάθε πύλης είναι γνωστή αλλιώς εκτιμάται μία εικονική πύλη για την συνάρτηση (πχ από τον αριθμό των μεταβλητών της συνάρτησης.
- ✓ Η καλωδίωση είναι ανάλογη της συνολικής επιφάνειας των πυλών.
- ✓ Υπάρχει σχετική δυσκολία στην εκτίμηση της επιφάνειας από τις μεταβλητές γιατί απαιτείται βέλτιστη παραγοντοποίηση των εκφράσεων.

Βελτιστοποίηση Λογικών Δικτύων

<u>Χρονική βελτιστοποίηση</u>

- ✓ Είναι η ελαχιστοποίηση της καθυστέρησης του κρίσιμου μονοπατιού.
- ✓ Απαιτείται ο υπολογισμός δύο παραγόντων: της καθυστέρησης διάδοσης κάθε κορυφής και κάθε μονοπατιού.
- ✓ Η καθυστέρηση κορυφής μπορεί να υπολογιστεί από την βιβλιοθήκη πυλών ή να εκτιμηθεί από την εικονική πύλη.
- ✓ Ένας χονδρικός τρόπος εκτίμησης είναι η θεώρηση ενός unit καθυστέρησης σε κάθε πύλη.
- ✓ Ο υπολογισμός της καθυστέρησης ενός μονοπατιού είναι η εύρεση ορίου χειρότερης περίπτωσης στην διάδοση ενός γεγονότος.
- ✓ Μία απλή προσέγγιση είναι το άθροισμα των καθυστερήσεων όλων των κορυφών του μονοπατιού.

Βελτιστοποίηση Λογικών Δικτύων

- ✓ Πρέπει να γίνεται ανίχνευση των Λανθανόντων μονοπατιών (δεν μπορούν να διαδόσουν γεγονότα).
- ✓ Για την εκτίμηση των καλωδιώσεων χρησιμοποιούνται στατιστικά μοντέλα.
- ✓ Βηματική βελτιστοποίηση δικτύου με εφαρμογή λογικών μετασχηματισμών που διατηρούν την συμπεριφορά εισόδου/εξόδου.
- ✓ Οι λογικοί μετασχηματισμοί ορίζονται έτσι ώστε να διατηρούν την συμπεριφορά του κυκλώματος χωρίς να απαιτείται ο έλεγχός τους.
- ✓ Η σειρά εφαρμογής των μετασχηματισμών δεν επηρεάζει την συμπεριφορά.

Μετασχηματισμοί Λογικών Δικτύων

- ✓Οι λογικοί μετασχηματισμοί μπορούν να έχουν τοπική ή ολική επίδραση:

  (α) Τοπική: μετατροπή μίας τοπικής συνάρτησης χωρίς την αλλαγή της δομής.

  (β) Ολική: επηρεάζει την δομή του κυκλώματος με δημιουργία/διαγραφή ακμών-κορυφών.

- ✓Θεωρούμε ότι οι τοπικές συναρτήσεις περιγράφονται με Boolean εκφράσεις πιθανώς παραγοντοποιημένες.

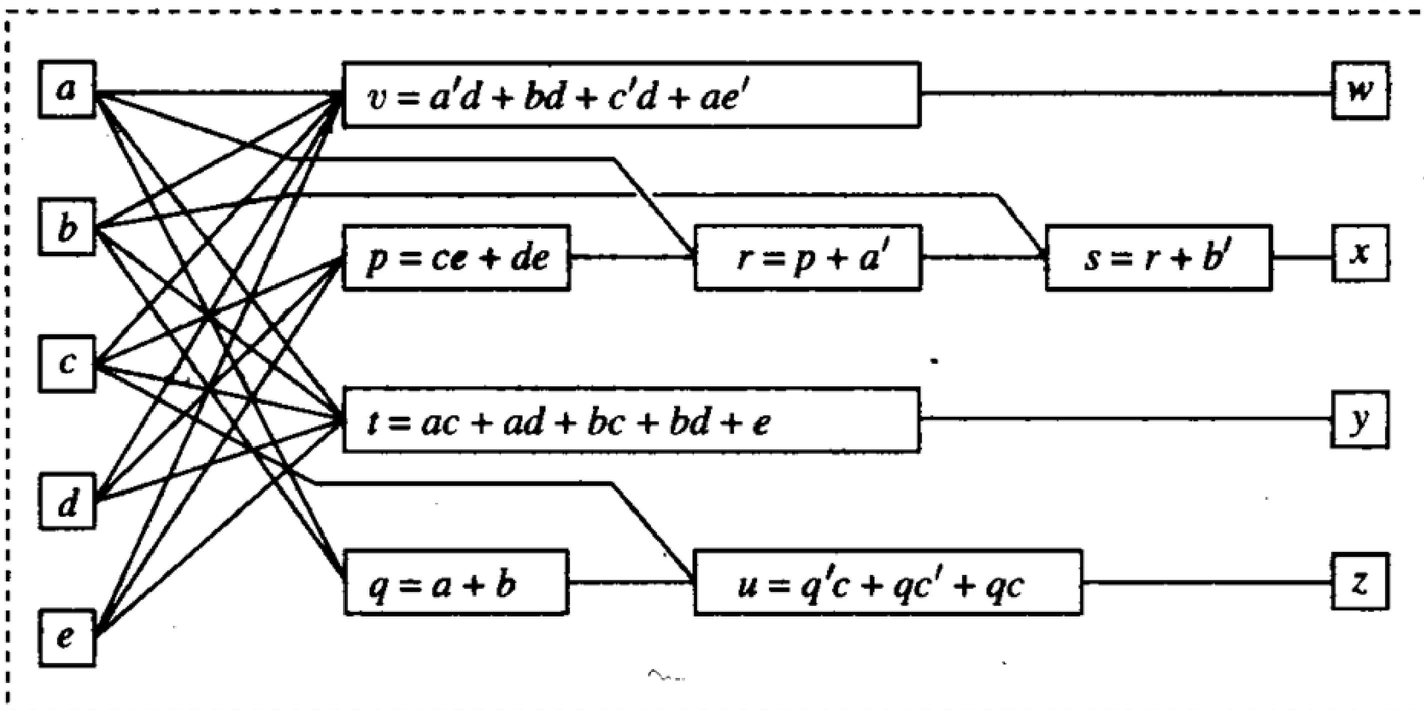
Μετασχηματισμοί Λογικών Δικτύων

Elimination.

- ✓ Είναι η αφαίρεση μίας κορυφής από το δίκτυο.
- ✓ Η μεταβλητή που αντιστοιχεί σε αυτήν την κορυφή αντικαθιστάται από την αντίστοιχη έκφραση σε κάθε εμφάνισή της στο δίκτυο.

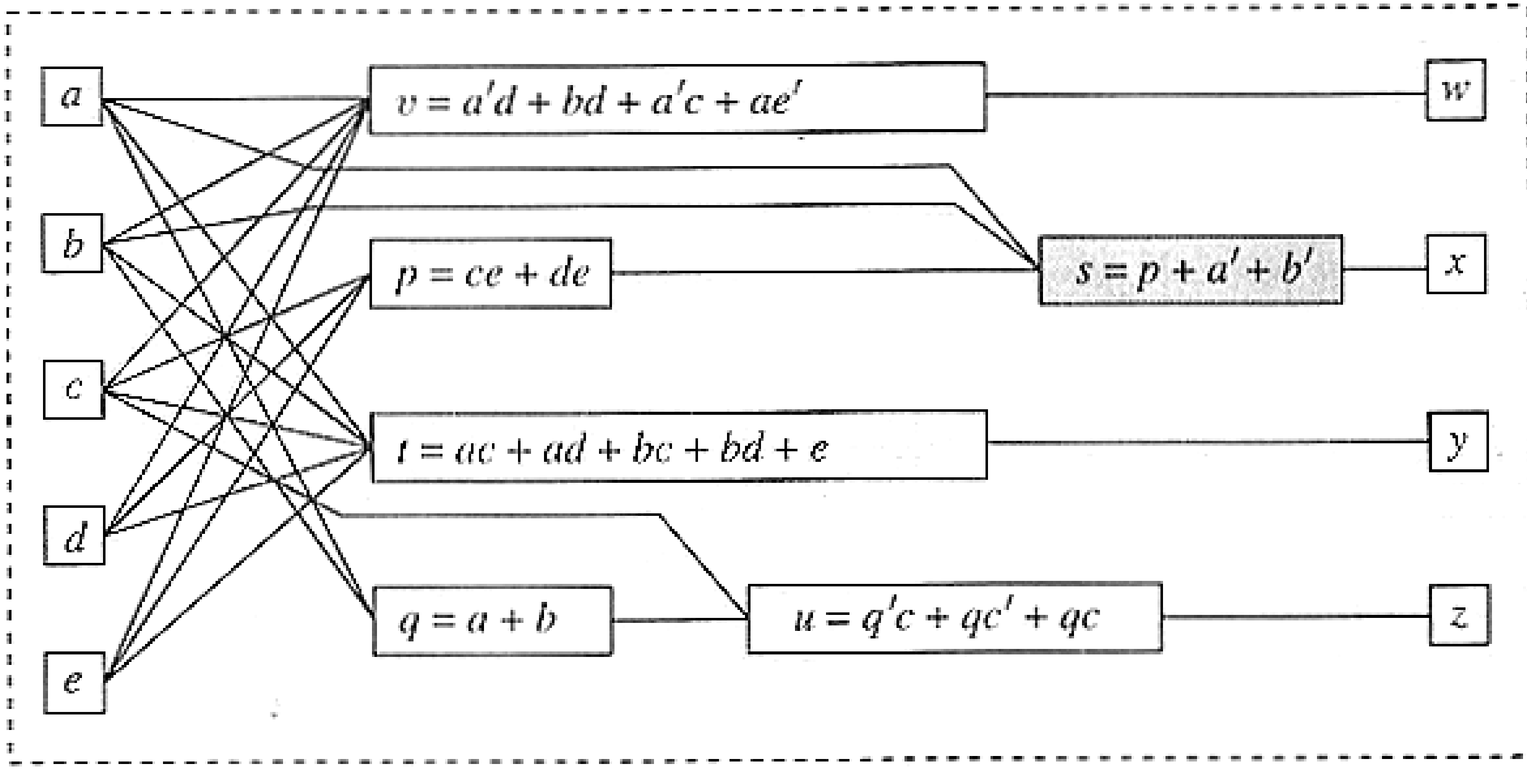
Πχ. Η αφαίρεση της κορυφής v_r οδηγεί στην αντικατάσταση του r με το $p+a'$

Μετασχηματισμοί Λογικών Δικτύων



(a)

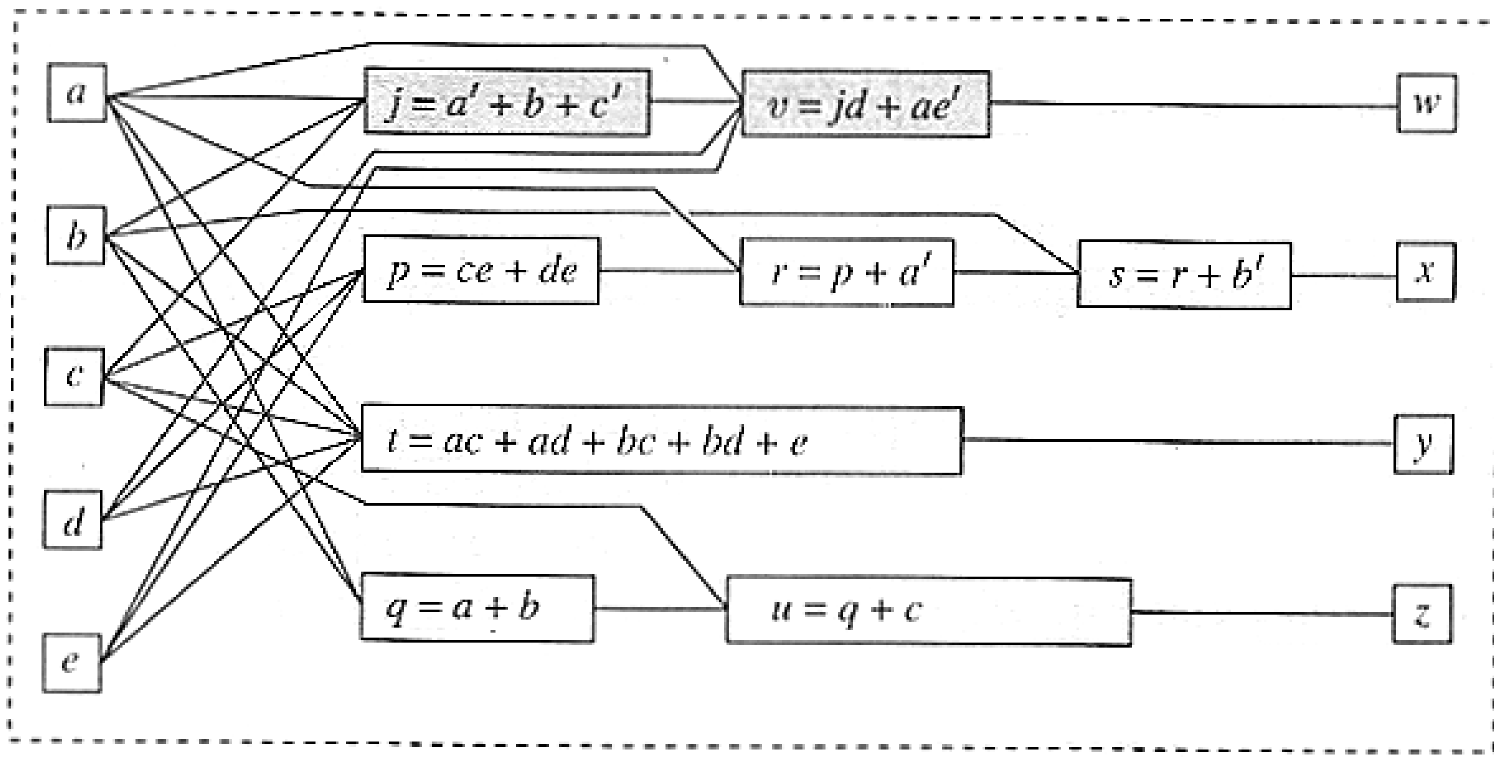
Μετασχηματισμοί Λογικών Δικτύων



- Αφαίρεσης μίας κορυφής που αντιστοιχεί σε απλή τοπική συνάρτηση, η οποία μπορεί να διαδοθεί σε άλλες τοπικές συναρτήσεις.

Μετασχηματισμοί Λογικών Δικτύων

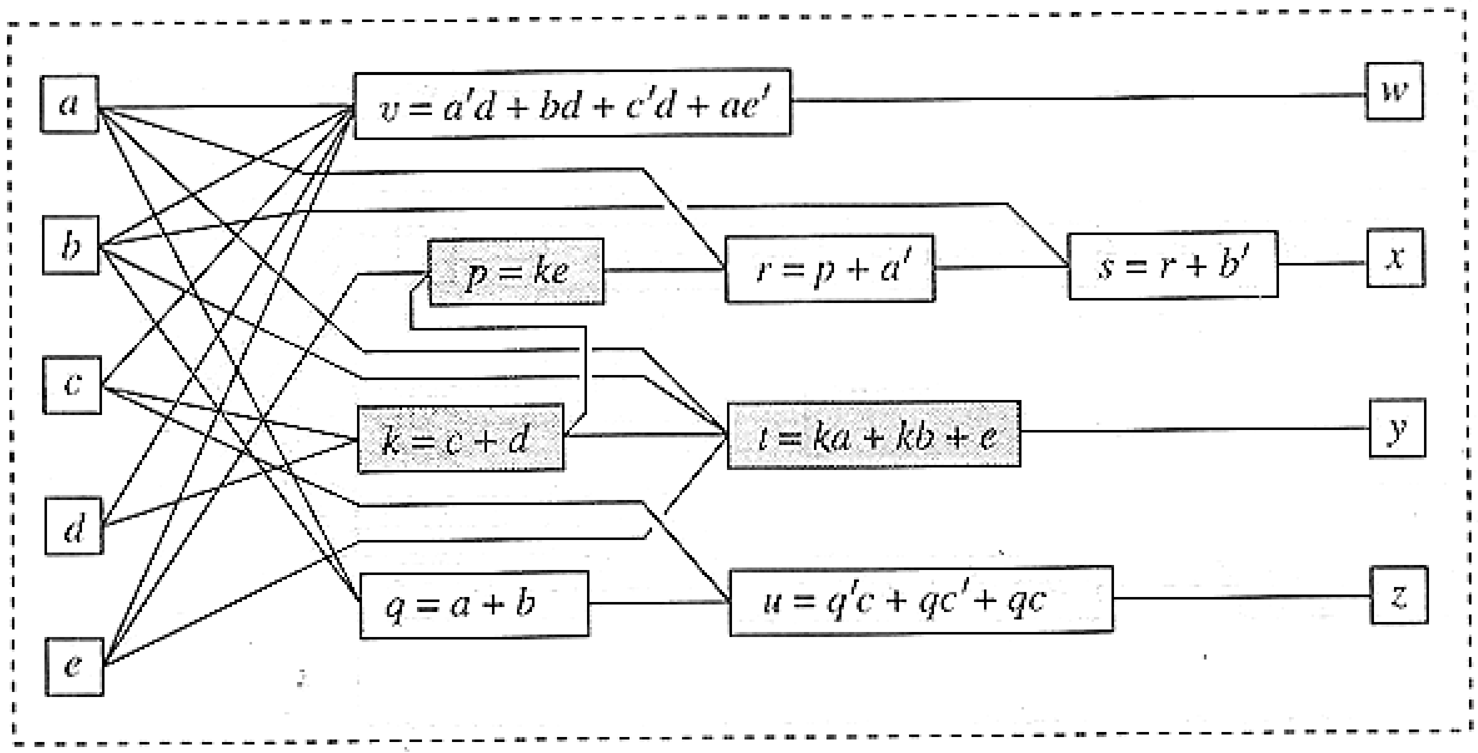
Decomposition. Είναι η αντικατάσταση μίας εσωτερικής κορυφής με δύο ή περισσότερες κορυφές που δομούν ένα υποδίκτυο ισοδύναμο στην αρχική κορυφή. Πχ. $f_v = a'd+bd+c'd+ae' = (a'+b+c')d+ae' = jd+ae'$ όπου $j = a'+b+c'$.



Ένας λόγος χρήσης του μετασχηματισμού είναι η αντικατάσταση μίας περίπλοκης συνάρτησης σε δύο ή περισσότερες απλούστερες συναρτήσεις.

Μετασχηματισμοί Λογικών Δικτύων

Extraction. Μία κοινή υπο-έκφραση δύο συναρτήσεων διαφορετικών κορυφών μπορεί να εξαχθεί και να δημιουργήσει μία τρίτη κορυφή η οποία απλοποιεί τις δύο συναρτήσεις.Πχ. p=(c+d)e, t=(c+d)(a+b)+e και θέτουμε k=c+d, p=ke, t=ka+kb+e



Ένας λόγος χρήσης του μετασχηματισμού είναι η απλοποίηση του δικτύου με εύρεση κοινών όρων.

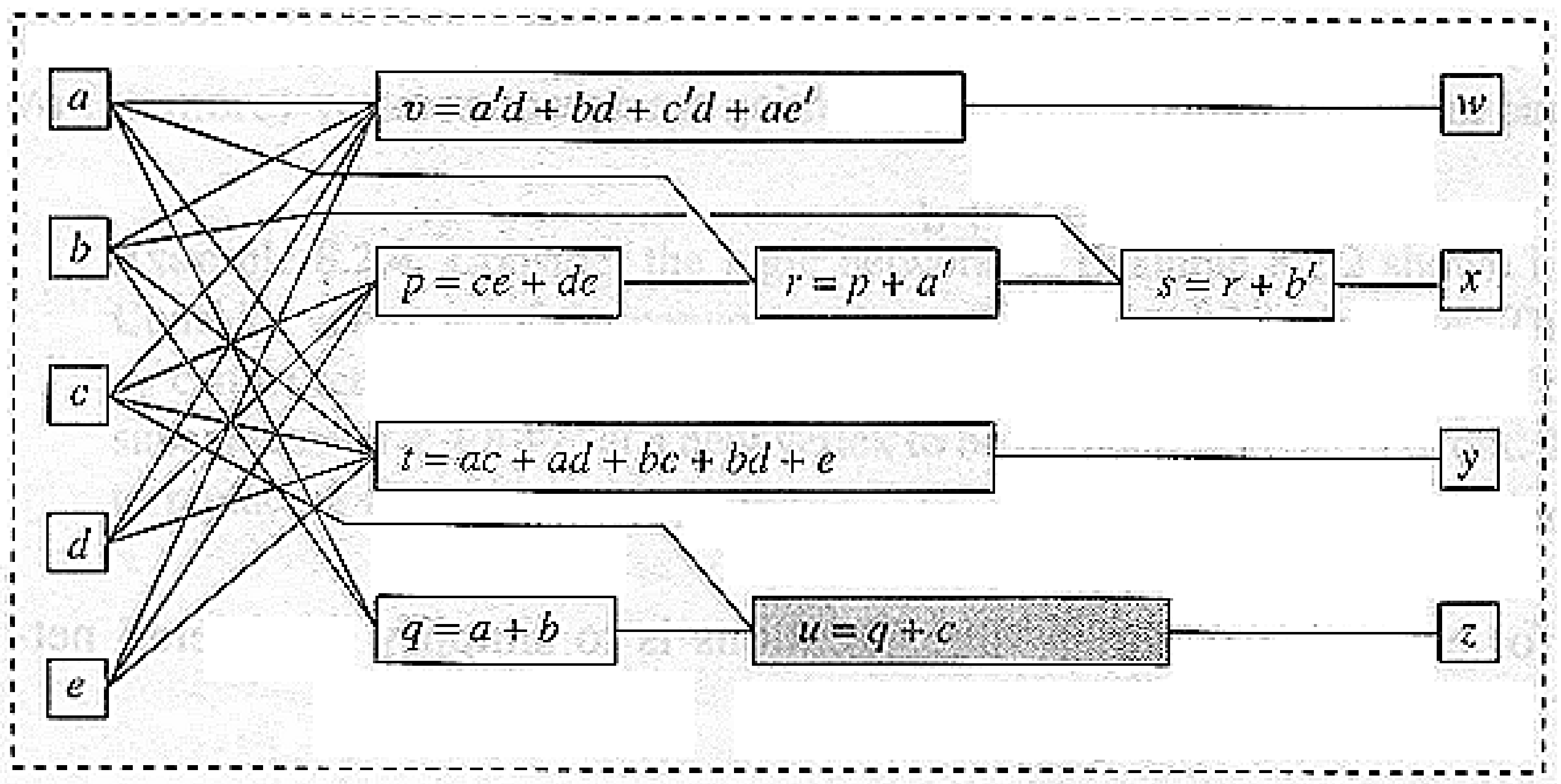
Μετασχηματισμοί Λογικών Δικτύων

Simplification.

- Είναι η μείωση της πολυπλοκότητας μίας συνάρτησης με την απλοποίηση της.
- Εάν δεν αλλάζει το σύνολο εισόδων τοπικά της συνάρτησης τότε πρόκειται για τοπικό μετασχηματισμό.
- Εάν το σύνολο εισόδων αλλάξει τότε αλλάζει η δομή του κυκλώματος και ο μετασχηματισμός είναι ολικός.

Πχ. $f_u=q'c+qc'+qc$, η οποία απλοποιείται στην $f_u=q+c$

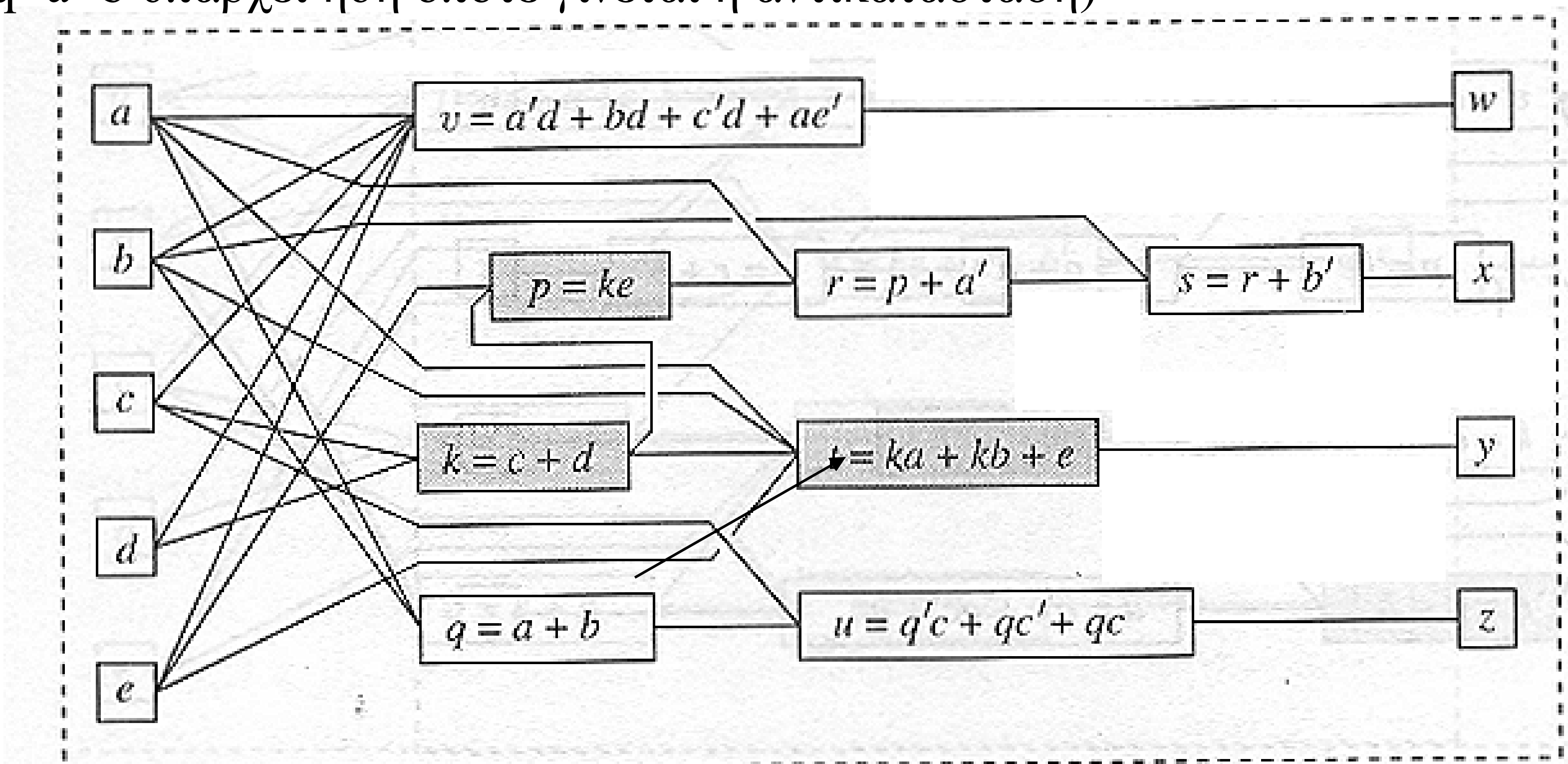
Μετασχηματισμοί Λογικών Δικτύων



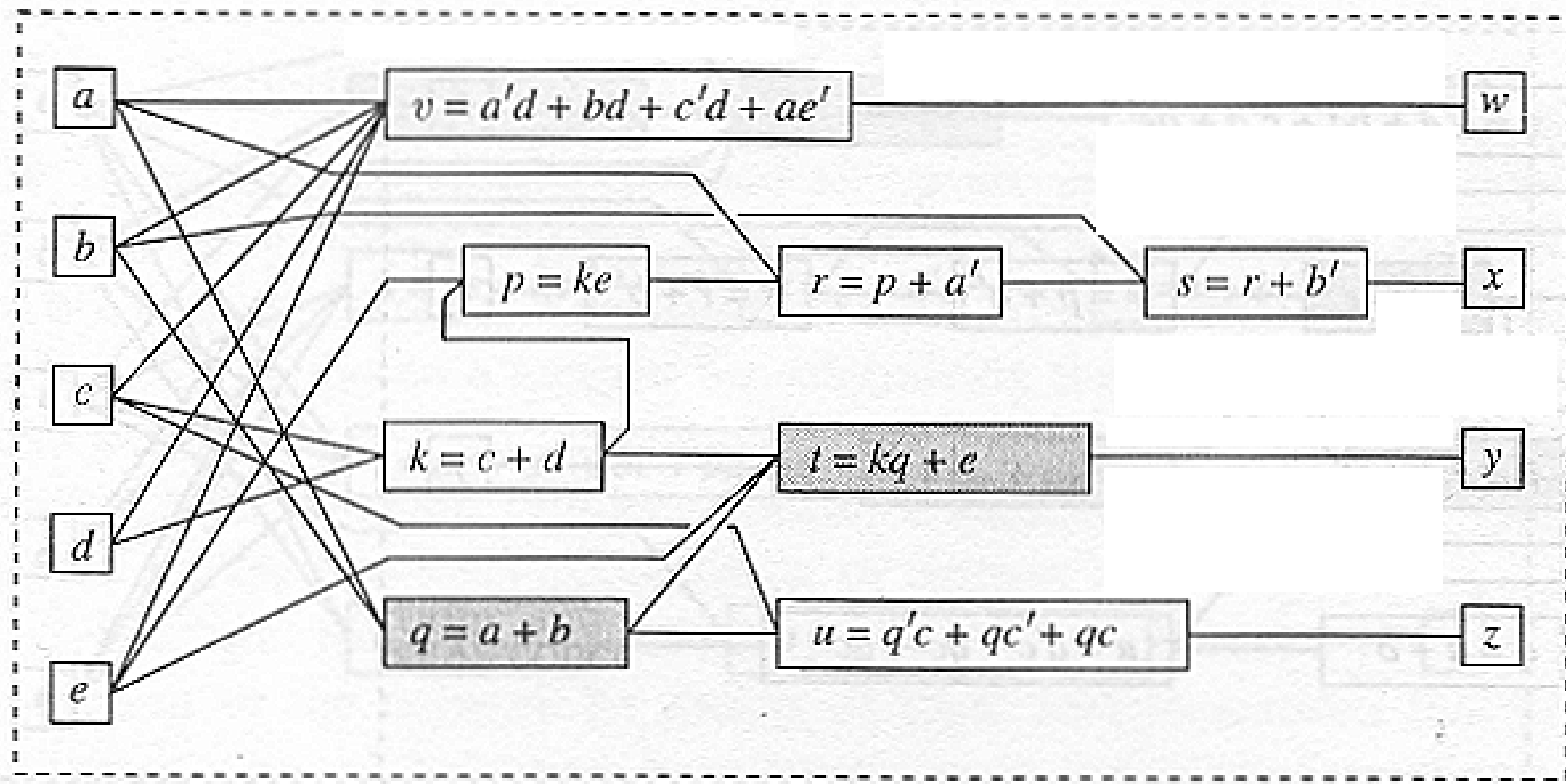
Μετασχηματισμοί Λογικών Δικτύων

Substitution. Μία συνάρτηση μειώνεται σε πολυπλοκότητα με την χρήση πρόσθετων εισόδων. Ο μετασχηματισμός οδηγεί σε νέες εξαρτήσεις αλλά μπορεί να καταστρέψει άλλες.

Πχ. f_t=ka+kb+e, η οποία απλοποιείται στην f_t=kq+e με q=a+b. (προσοχή: το q=a+b υπάρχει ήδη οπότε γίνεται η αντικατάσταση)



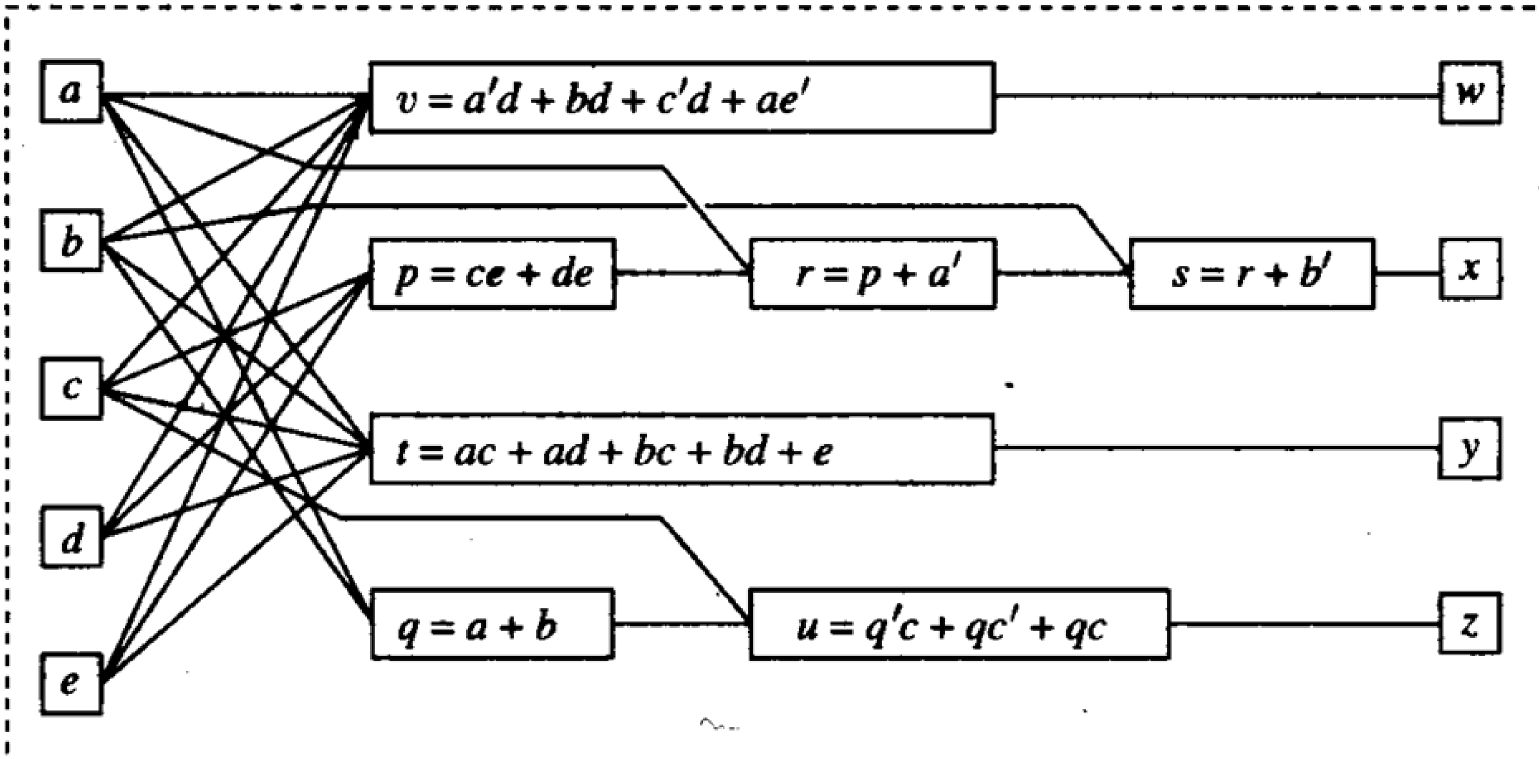
Μετασχηματισμοί Λογικών Δικτύων



Μετασχηματισμοί Λογικών Δικτύων

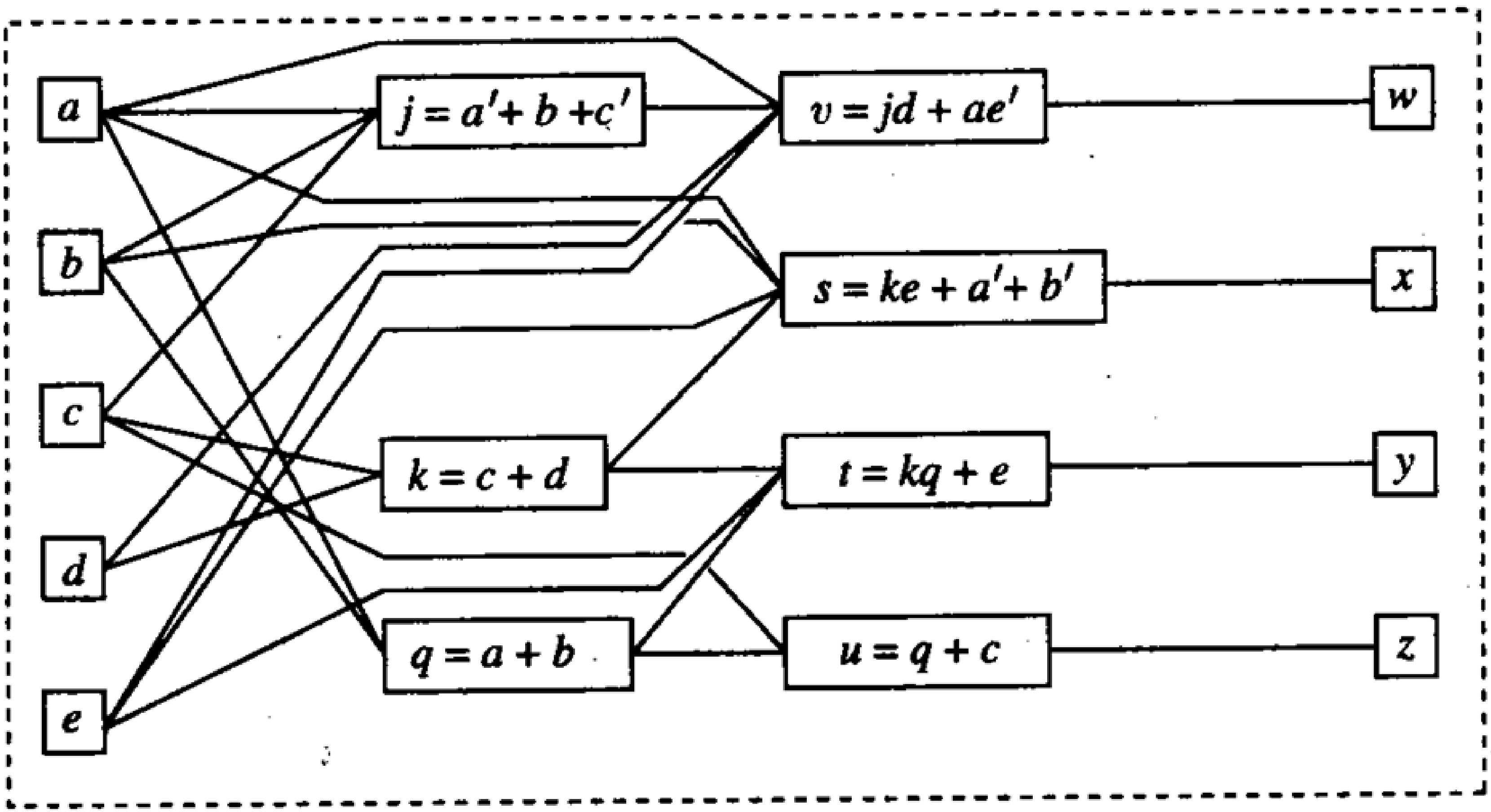
Μία λογική κάλυψη είναι ελάχιστη (τοπικά) όταν οι διαθέσιμοι τελεστές δεν μπορούν να την βελτιώσουν παραπέρα.

Για την επίτευξη ελάχιστης κάλυψης εφαρμόζονται επαναληπτικά οι μετασχηματισμοί.



(a)

Μετασχηματισμοί Λογικών Δικτύων



Προσεγγίσεις Βελτιστοποίησης

Υπάρχουν δύο βασικοί περιορισμοί στην βελτιστοποίηση δικτύων με λογικούς μετασχηματισμούς:

1. Με δεδομένο ένα σύνολο μετασχηματισμών είναι δύσκολο να ισχυριστούμε ότι ένα ισοδύναμο δίκτυο μπορεί να δημιουργηθεί από μία ακολουθία αυτών. Έτσι η βέλτιστη λύση ή έστω οι βιώσιμες λύσεις μπορεί να μην είναι εφικτές.

2. Διαφορετικές ακολουθίες μετασχηματισμών οδηγούν σε διαφορετικά αποτελέσματα που μπορεί να αντιστοιχούν σε διαφορετικά τοπικά βέλτιστα.

Όλες οι ευριστικές προσεγγίσεις χρησιμοποιούν τους μετασχηματισμούς και πετυχαίνουν βηματικά την βελτίωση του δικτύου.

Η κύρια κατηγοριοποίηση των μεθόδων είναι: α) αλγοριθμικές και β) βασισμένες σε κανόνες.

Αλγοριθμική Προσέγγιση

<u>Αλγοριθμικές μέθοδοι</u>

- ✓ Καθορισμός ενός αλγορίθμου για κάθε είδος μετασχηματισμού.
- ✓ Ο αλγόριθμος ανιχνεύει που και πως μπορεί να εφαρμοστεί και σταματά όταν δεν μπορούν να γίνουν πλέον ωφέλιμοι μετασχηματισμοί.

(Παραδείγματα: MIS, BOLD).

Πλεονέκτημα αλγοριθμικής μεθόδου:

Μετασχηματισμοί ενός τύπου εφαρμόζονται συστηματικά και μπορούν να οδηγήσουν σε κάποιες ιδιότητες του κυκλώματος (πχ πρωταρχικότητα και μη-πλεονασμός).

Προσέγγιση Κανόνων

Βασισμένες σε κανόνες μέθοδοι

- ✓ Οι μετασχηματισμοί διαφορετικών τύπων μπορούν να εναλλαχθούν σύμφωνα με ένα σύνολο κανόνων
- ✓ Το σύνολο κανόνων μιμείται τα βήματα βελτιστοποίησης ενός σχεδιαστή.
- ✓ Στην βάση δεδομένων των κανόνων αποθηκεύονται ζεύγη διανυσμάτων.
- ✓ Το δεύτερο διάνυσμα αντιπροσωπεύει μία καλύτερη υλοποίηση του πρώτου.
- ✓ Γίνεται ανίχνευση των υποδικτύων που αντιστοιχούν στο πρώτο και αντικαθιστούνται από το υποδίκτυο που αντιστοιχεί στο δεύτερο διάνυσμα

(Παράδειγμα: IBM Logic Synthesis System – LSS).

Αλγοριθμική Προσέγγιση

Παράδειγμα: Αλγόριθμος Elimination

Επαναληπτική εξάλειψη μεταβλητών (κορυφών) για την μείωση των βαθμίδων ενός δικτύου.

Eliminate (G_n(V, E), k) {
  repeat {
    v_x=selected vertex with value not larger than k;
    if (v_x=0) return;
    Replace x by f_x in the network } }

Αλγοριθμική Προσέγγιση

- ✓ Η τιμή που σχετίζεται με κάθε κορυφή είναι η αύξηση σε επιφάνεια ή καθυστέρηση που οφείλεται στην διαγραφή της μεταβλητής.
- ✓ Για να μην έχουμε αύξηση σε επιφάνεια ή καθυστέρηση η τιμή κατωφλίου k πρέπει να είναι ίση με 0.
- ✓ Με μικρή τιμή του k έχουμε μείωση των βαθμίδων με μικρή αύξηση της επιφάνειας ή της καθυστέρησης.
- ✓ Μεγάλη τιμή του k επιτρέπει την μη περιορισμένη μείωση βαθμίδων.
- ✓ Η επιλογή των βαθμίδων γίνεται greedy.
- ✓ Οι υποψήφιες κορυφές πρέπει να είναι εσωτερικές (όχι άμεσα παρατηρήσιμες από τις εξόδους).

Αλγοριθμική Προσέγγιση

~~p = ce + de~~
q = a + b
~~r = p + a΄~~
s = r + b΄
t = ac+ad+bc+bd+e
u = q΄c+qc΄+qc
v = a΄d+bd+c΄d+ae΄
w = v
x = s
y = t
z = u

q = a + b
s = ce + de + a΄ + b΄
t = ac+ad+bc+bd+e
u = q΄c+qc΄+qc
v = a΄d+bd+c΄d+ae΄
w = v
x = s
y = t
z = u

Κριτήριο: αριθμός των literals

Κατώφλι: 0

Αντικατάσταση των κορυφών v_r, v_p

Αλγοριθμική Προσέγγιση

- ✓ Ο αλγόριθμος elimination μειώνει τον αριθμό των βαθμίδων και καταστρέφει την δομή του δικτύου.
- ✓ Το αντίθετο πρόβλημα, δηλαδή η αύξηση των βαθμίδων ή των εξαρτήσεων είναι πολύ πιο δύσκολο πρόβλημα (παράδειγμα οι μετασχηματισμοί extraction & substitution που ψάχνουν για κοινές υποεκφράσεις).
- ✓ Η δυσκολία των μεθόδων που προσθέτουν τμήματα οφείλεται στους βαθμούς ελευθερίας των συναρτήσεων Boole.