Προβληματα Γενεσης: Αλγοριθμοι και Υπολογιστικη Πολυπλοκοτητα

Ηλίας Κ. Σταυρόπουλος

Ιούνιος 2003

Πανεπιστήμιο Πατρών – Πολυτεχνική Σχολή
Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

Περίληψη

Σε αρκετά πρακτικά προβλήματα δεν αρκεί η γνώση της ύπαρξης μιας λύσης ή στην εύρεση αυτής, αλλά απαιτείται η εύρεση και καταγραφή όλων των δυνατών λύσεων. Τα προβλήματα στα οποία ζητείται η εύρεση όλων των δυνατών δομών που ικανοποιούν συγκεκριμένες προδιαγραφές και ιδιότητες καλούνται προβλήματα γένεσης. Σε ένα πρόβλημα γένεσης είναι πιθανό οι δυνατές λύσεις να είναι εκθετικά πολλές, με άμεση συνέπεια οποιαδήποτε τεχνική (αλγόριθμος) εύρεσης αυτών να απαιτεί εκθετικό χρόνο αναφορικά με το μέγεθος της εισόδου του προβλήματος. Έτσι, για τη μελέτη της υπολογιστικής πολυπλοκότητας των προβλήματων γένεσης και τη μέτρηση της αποδοτικότητας των αλγορίθμων που επιλύουν τέτοια προβλήματα χρησιμοποιούνται κατάλληλες μετρικές που λαμβάνουν υπόψη τους όχι μόνο το μέγεθος της εισόδου αλλά και το μέγεθος της εξόδου του προβλήματος.

Στην Επιστήμη των Υπολογιστών αρκετές είναι οι περιπτώσεις όπου ζητούνται όχι όλες οι λύσεις αλλά οι πιο χαρακτηριστικές ή οι πιο φυσικές. Για παράδειγμα, υπάρχουν συγκεκριμένα προβλήματα στη Λογική, στην Τεχνητή Νοημοσύνη, στην Εξόρυξη Δεδομένων και στα Κατανεμημένα Συστήματα, όπου ζητείται η εύρεση ακραίων δομών (μεγιστοτικών ή ελαχιστοτικών). Η εύρεση των ακραίων δομών έχει ιδιαίτερη σημασία αφού αυτές αποτελούν το σύνολο μεταξύ των δομών που ικανοποιούν τις προδιαγραφές του προβλήματος και αυτών που δεν τις ικανοποιούν. Σε αρκετές περιπτώσεις, ο πιο κατάλληλος τρόπος για να μπορέσει κανείς να κατανοήσει αλλά και εκφράσει αυτές τις έννοιες είναι μέσω της διατέμνουσας ενός υπεργραφήματος. Ένα υπεργράφημα $\mathcal{H}$ είναι ένα γενικευμένο γράφημα με κάθε υπερακμή του να περιέχει περισσότερους από δύο κόμβους. Σε αναλογία, μια διατέμνουσα είναι ότι και το κάλυμμα κορυφών στα γραφήματα: ένα σύνολο κόμβων που έχει μη κενή τομή με κάθε υπερακμή. Μια διατέμνουσα είναι ελαχιστοτική εάν δεν είναι υπερσύνολο καμίας άλλης διατέμνουσας. Το σύνολο όλων των

ελαχιστοτικών διατεμνουσών ενός υπεργραφήματος $\mathcal{H}$ αποτελεί το διατέμνον υπεργράφημα $Tr(\mathcal{H})$ του $\mathcal{H}$.

Η ΓΕΝΕΣΗ ΔΙΑΤΕΜΝΟΝΤΟΣ ΥΠΕΡΓΡΑΦΗΜΑΤΟΣ αποτελεί ένα από τα πιο ενδιαφέροντα προβλήματα στη Θεωρία Υπεργραφημάτων. Εκτός από τον μεγάλο αριθμό εμφανίσεων του ως κεντρικό υποπρόβλημα σε διάφορες περιοχές της Επιστήμης των Υπολογιστών, αυτό που επίσης προσδίδει ενδιαφέρον στη ΓΕΝΕΣΗ ΔΙΑΤΕΜΝΟΝΤΟΣ ΥΠΕΡΓΡΑΦΗΜΑΤΟΣ είναι η ύπαρξη προβλημάτων που δεν είναι τίποτε άλλο παρά μία διαφορετική διατύπωση αυτού. Τέτοια προβλήματα είναι η εύρεση των ελαχιστοτικών όρων της δυϊκής μιας μονότονης λογικής συνάρτησης σε διαζευκτική κανονική μορφή (ΓΕΝΕΣΗ ΔΥΪΚΗΣ ΣΥΝΑΡΤΗΣΗΣ) και η εύρεση των μεγιστοτικών μοντέλων μιας λογικής έκφρασης σε συζευκτική κανονική μορφή (ΓΕΝΕΣΗ ΜΕΓΙΣΤΟΤΙΚΩΝ ΜΟΝΤΕΛΩΝ) όπου κάθε πρόταση της είναι πλήρως αρνητική. Η ΓΕΝΕΣΗ ΔΙΑΤΕΜΝΟΝΤΟΣ ΥΠΕΡΓΡΑΦΗΜΑΤΟΣ είναι πολυωνυμικά ισοδύναμη με τη ΓΕΝΕΣΗ ΔΥΪΚΗΣ ΣΥΝΑΡΤΗΣΗΣ και η ΓΕΝΕΣΗ ΜΕΓΙΣΤΟΤΙΚΩΝ ΜΟΝΤΕΛΩΝ είναι τουλάχιστον τόσο δύσκολη όσο η ΓΕΝΕΣΗ ΔΙΑΤΕΜΝΟΝΤΟΣ ΥΠΕΡΓΡΑΦΗΜΑΤΟΣ.

Η μελέτη της υπολογιστικής πολυπλοκότητας των παραπάνω προβλημάτων γένεσης ακραίων δομών έχει προσελκύσει τις τελευταίες δεκαετίες το ερευνητικό ενδιαφέρον αρκετών ερευνητών. Παρά τις έντονες προσπάθειες, εξακολουθεί να παραμένει αναπάντητο το ερώτημα της ύπαρξης ή όχι αλγορίθμου πολυωνυμικού στην έξοδο (δηλαδή, αλγορίθμου που ο χρόνος εκτέλεσης του φράσσεται από πολυώνυμο στο μέγεθος της εισόδου και της εξόδου) για τη ΓΕΝΕΣΗ ΔΙΑΤΕΜΝΟΝΤΟΣ ΥΠΕΡΓΡΑΦΗΜΑΤΟΣ. Η πολυπλοκότητα του προβλήματος αυτού εξαρτάται άμεσα από την αποφαντική του έκδοση ΔΙΑΤΕΜΝΟΝ ΥΠΕΡΓΡΑΦΗΜΑ. Το ΔΙΑΤΕΜΝΟΝ ΥΠΕΡΓΡΑΦΗΜΑ ανήκει στην κλάση **coNP** ενώ σε αρκετές περιπτώσεις μπορεί να λυθεί πολυωνυμικά. Το 1996 οι Fredman και Khachiyan παρουσίασαν έναν αλγόριθμο που επιλύει την ΔΥΪΚΟΤΗΤΑ (ισοδύναμο πρόβλημα με το ΔΙΑΤΕΜΝΟΝ ΥΠΕΡΓΡΑΦΗΜΑ) σε χρόνο υποεκθετικό $n^{o(\log n)}$, όπου n είναι το μέγεθος της εισόδου του προβλήματος. Το αποτέλεσμα αυτό φανέρωσε ότι το ΔΙΑΤΕΜΝΟΝ ΥΠΕΡΓΡΑΦΗΜΑ δεν είναι πλήρες για την κλάση **coNP**, εκτός εάν κάθε **coNP**-πλήρες πρόβλημα μπορεί να λυθεί σε υποεκθετικό χρόνο, αλλά ανήκει μάλλον σε μια ενδιάμεση κλάση μεταξύ των κλάσεων **P** και **coNP**. Ο αλγόριθμος των Fredman και Khachiyan μπορεί θεωρητικά να χρησιμοποιηθεί για τη γένεση του διατέμνοντος υπεργραφήματος σε αυξητικά υποεκθετικό χρόνο.

Από τα παραπάνω γίνεται φανερό ότι τα προβλήματα γένεσης που περιγράψαμε (κεντρικά προβλήματα σε πολλές περιοχές της Επιστήμης των Υπολογιστων) δεν έχουν κατανοηθεί πλήρως και η ακριβής πολυπλοκότητα τους παραμένει ανοικτό πρόβλημα. Αντικείμενο της παρούσας διδακτορικής διατριβής είναι η μελέτη της υπολογιστικής πολυπλοκότητας των προβλημάτων αυτών και ο σχεδιασμός κατάλληλων αλγορίθμων για την αποδοτική επίλυση τους.

Ξεκινάμε δίνοντας τους ορισμούς των κατάλληλων μετρικών για τη μέτρηση της αποδοτικότητας των αλγορίθμων επίλυσης προβλημάτων με μεγάλη έξοδο. Στη συνέχεια διατυπώνουμε αυστηρά τα προβλήματα γένεσης που θα μας απασχολήσουν και δίνουμε μια σύντομη περιγραφή του τρόπου με τον οποίο τα παραπάνω προβλήματα εμφανίζονται ως κεντρικό υποπρόβλημα σε διάφορες περιοχές της Επιστήμης των Υπολογιστών, κάνοντας εκτεταμένη αναφορά στην βιβλιογραφία.

Για την αλγοριθμική προσέγγιση του προβλήματος ΓΕΝΕΣΗ ΔΙΑΤΕΜΝΟΝΤΟΣ ΥΠΕΡΓΡΑΦΗΜΑΤΟΣ, παρουσιάζουμε έναν πρακτικό αλγόριθμο να παράγει τις ελαχιστοτικές διατέμνουσες ενός υπεργραφήματος γρήγορα και αποδοτικά και είναι κατάλληλος για στιγμιότυπα μεγάλου μεγέθους. Ο αλγόριθμος μας ενσωματώνει μια σειρά νέων ιδεών όπως τη χρήση γενικευμένων κόμβων, τον κατά βάθος υπολογισμό των ελαχιστοτικών διατεμνουσών και την επιλεκτική παραγωγή της επόμενης ελαχιστοτικής διατέμνουσας μέσω της έννοιας των κατάλληλων κόμβων. Αποδεικνύουμε ότι ο προτεινόμενος αλγόριθμος παράγει ορθά την έξοδο χωρίς επαναγένεση. Αν και μέχρι στιγμής δεν έχουμε αποδείξει κάποιο άνω φράγμα για την χρονική πολυπλοκότητα του αλγορίθμου, αποδεικνύουμε εντούτοις ότι ο αλγόριθμος απαιτεί χώρο πολυωνυμικό στο μέγεθος του υπεργραφήματος εισόδου. Παράλληλα, η πειραματική αποτίμηση του αλγορίθμου έδειξε πως ο αλγόριθμος παράγει την έξοδο με μια απροσδόκητη ομοιομορφία, αρκετά γρήγορα και αποδοτικά, ακόμα και για μεγάλα στιγμιότυπα.

Αναφορικά με τη ΓΕΝΕΣΗ ΜΕΓΙΣΤΟΤΙΚΩΝ ΜΟΝΤΕΛΩΝ, παρουσιάζουμε μια διαδικασία επίλυσης η οποία απαλοίφει τις θετικές εμφανίσεις των μεταβλητών μιας γενικής λογικής έκφρασης σε συζευκτική κανονική μορφή και την μετασχηματίζει σε πλήρως αρνητική. Αποδεικνύουμε ότι η τελική λογική έκφραση έχει το ίδιο σύνολο μεγιστοτικών μοντέλων με την αρχική, οπότε η γένεση των μεγιστοτικών μοντέλων αυτής μπορεί να γίνει με εφαρμογή οποιουδήποτε αλγορίθμου γένεσης ελαχιστοτικών διατεμνουσών. Ωστόσο, η τελική πλήρως αρνητική έκφραση ενδέχεται να είναι εκθετικά μεγαλύτερη από την αρχική, γεγονός που δεν οφείλεται στην προτεινόμενη διαδικασία επίλυσης αφού, όπως αποδεικνύουμε, ακόμα και στην περίπτωση των Horn λογικών εκφράσεων (κάθε πρόταση περιέχει το πολύ μία θετική μεταβλητή), η μικρότερη ισοδύναμη ως προς τα μεγιστοτικά μοντέλα πλήρως αρνητική έκφραση είναι εκθετικά μεγαλύτερη από την αρχική. Αποδεικνύουμε ότι η γένεση των μεγιστοτικών μοντέλων μπορεί να γίνει σε αυξητικά υποεκθετικό χρόνο στην περίπτωση που το πλήθος των μεταβλητών με θετική εμφάνιση είναι σταθερό και σε αυξητικά $O(m^{poly(\log m)})$ χρόνο στην περίπτωση που το πλήθος των μεταβλητών με θετική εμφάνιση είναι $O(\log\log m)$, όπου m είναι το πλήθος των προτάσεων της έκφρασης. Χρησιμοποιώντας την προτεινόμενη διαδικασία επίλυσης, παρουσιάζουμε έναν αλγόριθμο που παράγει τα μεγιστοτικά μοντέλα μιας λογικής έκφρασης σε 2-ΣΚΜ (κάθε πρόταση περιέχει το πολύ δύο προσημασμένες μεταβλητές) με πολυωνυμική καθυστέρηση.

Η μελέτη της υπολογιστικής πολυπλοκότητας του προβλήματος γένεσης μεγιστοτικών μοντέλων μιας λογικής έκφρασης ξεκινά από απλούστερα αλλά εξίσου δύσκολα προβλήματα. Αποδεικνύουμε ότι το πρόβλημα του ελέγχου εάν ένα μοντέλο είναι μεγιστοτικό είναι **NP**-πλήρες για γενικές λογικές εκφράσεις ενώ μπορεί να λυθεί πολυωνυμικά για τις πολυωνυμικές περιπτώσεις της ικανοποιησιμότητας. Για γενικές λογικές εκφράσεις **NP**-δύσκολο είναι και το πρόβλημα του ελέγχου της ύπαρξης ενός ακόμα μεγιστοτικού μοντέλου, δοθέντων κάποιων μεγιστοτικών μοντέλων αυτών. Ιδιαίτερο ενδιαφέρον παρουσιάζουν οι πολυωνυμικές περιπτώσεις του προβλήματος της ικανοποιησιμότητας, αφού στις υπόλοιπες περιπτώσεις η ΓΕΝΕΣΗ ΜΕΓΙΣΤΟΤΙΚΩΝ ΜΟΝΤΕΛΩΝ είναι προφανώς δύσκολη. Αποδεικνύουμε ότι το πρόβλημα του ελέγχου της ύπαρξης ενός ακόμα μεγιστοτικού μοντέλου μιας Horn λογικής έκφρασης είναι **NP**-πλήρες και κατά συνέπεια δεν υπάρχει αλγόριθμος πολυωνυμικός στην έξοδο για τη γένεση των μεγιστοτικών μοντέλων μιας Horn λογικής έκφρασης, εκτός αν $\mathbf{P} = \mathbf{NP}$. Δείχνουμε επίσης ότι η ΓΕΝΕΣΗ ΜΕΓΙΣΤΟΤΙΚΩΝ ΜΟΝΤΕΛΩΝ επιλύεται αποδοτικά για λογικές εκφράσεις σε 2-ΣΚΜ χρησιμοποιώντας τον προτεινόμενο κανόνα επίλυσης και αφήνουμε ανοικτή την περίπτωση της γένεσης των μεγιστοτικών μοντέλων αφινικών λογικών εκφράσεων.

Για γενικές λογικές εκφράσεις η ΓΕΝΕΣΗ ΜΕΓΙΣΤΟΤΙΚΩΝ ΜΟΝΤΕΛΩΝ είναι τουλάχιστον τόσο δύσκολη όσο η ΓΕΝΕΣΗ ΔΙΑΤΕΜΝΟΝΤΟΣ ΥΠΕΡΓΡΑΦΗΜΑΤΟΣ. Η ύπαρξη αλγορίθμου πολυωνυμικού στην έξοδο για τη ΓΕΝΕΣΗ ΔΙΑΤΕΜΝΟΝΤΟΣ ΥΠΕΡΓΡΑΦΗΜΑΤΟΣ ισοδυναμεί με την ύπαρξη πολυωνυμικού αλγορίθμου για τη ΔΥΪΚΟΤΗΤΑ. Με βάση στον ντετερμινιστικό αλγόριθμο των Fredman και Khachiyan, αποδεικνύουμε ότι η ΔΥΪΚΟΤΗΤΑ μπορεί να επιλυθεί με περιορισμένο μη ντετερμινισμό. Παρουσιάζουμε έναν μη ντετερμινιστικό αλγόριθμο ο οποίος επιλύει τη ΔΥΪΚΟΤΗΤΑ σε πολυωνυμικό χρόνο κάνοντας $O(\log^2 n)$ μη ντετερμινιστικές επιλογές, όπου n είναι το μέγεθος της εισόδου. Το αποτέλεσμα αυτό κατατάσσει τη ΔΥΪΚΟΤΗΤΑ στην κλάση $\mathbf{coNP}[\log^2 n]$, την υποκλάση της $\mathbf{coNP}$ όπου μόνο τα πρώτα $O(\log^2 n)$ βήματα είναι μη ντετερμινιστικά. Δίνουμε έτσι ένα καλύτερο άνω φράγμα στην πολυπλοκότητα χρόνου του προβλήματος αυτού, αφήνοντας ανοικτό το πρόβλημα του υπολογισμού ενός καλύτερου κάτω φράγματος.

Generation Problems: Algorithms and Complexity

Elias C. Stavropoulos

June 2003

A Thesis submitted in conformity with the requirements
for the Degree of Doctor of Philosophy in the
Computer Engineering & Informatics Department,
University of Patras, Greece

Abstract

Generation problems are problems where finding all possible structures that satisfy certain specifications or properties is required. These problems emerge in many real world applications where the knowledge of the existence of a solution, or even finding the solution, is not sufficient. Since the number of all possible solutions may be exponentially large, the running time of any algorithm that solves a generation problem may be exponential with respect to the size of the input. Thus, for measuring the computational complexity of a generation problem as well as the efficiency of an algorithm for it, more elaborate complexity measures must be used that take into account not only the size of the input but the size of the output, too.

In Computer Science, one is often interested not in all solutions but in the most representative or natural ones. For example, there are certain problems in Logic, in Artificial Intelligence, in Data Mining, and in Distributed Systems where the interest is restricted in finding minimal or maximal (extremal) solutions. Finding extremal solutions is quite interesting since they act as a border between the structures that satisfy the specifications of the problem and those that do not. In many cases, a nice way of capturing the notion of minimal (or maximal) solution is by using the notion of the transversal of a hypergraph. A hypergraph $\mathcal{H}$ is a generalized graph where each hyperedge may have more than two nodes. A transversal (or, hitting set) of $\mathcal{H}$ is a set of nodes $\mathcal{T}$ that intersects every hyperedge of $\mathcal{H}$. A transversal $\mathcal{T}$ is minimal if no proper subset of $\mathcal{T}$ is a transversal of $\mathcal{H}$. The

transversal hypergraph $Tr(\mathcal{H})$ of $\mathcal{H}$ is the family of all minimal transversals of $\mathcal{H}$.

TRANSVERSAL HYPERGRAPH GENERATION is one of the most important problems in Graph Theory with a large number of applications in many areas of Computer Science. There are several cases where it appears as a central subproblem of many important problems, while there exist problems that are just a disguised form of the TRANSVERSAL HYPERGRAPH GENERATION. For example, generating all minimal transversal of a hypergraph is equivalent to generating all prime implicants of the dual form of a Boolean function in disjunctive normal form (DUAL FORM GENERATION). Moreover, TRANSVERSAL HYPERGRAPH GENERATION is equivalent to generating all maximal models of a Boolean expression in conjunctive normal form (MAXIMAL MODEL GENERATION) where all variables appear negated.

Complexity questions related to the above generation problems have been widely discussed in the literature during the last decades. However, the exact complexity of them is still an open issue. Although there are several algorithms that involve, in a more or less obvious way, the computation of transversal hypergraphs, the existence of an output-polynomial algorithm (that is, an algorithm that its running time is bounded by a polynomial of the combined size of the input and the output) is unknown. Its complexity is strongly depended on the complexity of its decision version TRANSVERSAL HYPERGRAPH. The TRANSVERSAL HYPERGRAPH problem is in its generality in **coNP**, while several polynomial cases exist. In 1996, Fredman and Khachiyan presented an algorithm that solves DUALIZATION (an equivalent problem with TRANSVERSAL HYPERGRAPH) in subexponential time $n^{o(\log n)}$, where n is the size of the input. This result implies that TRANSVERSAL HYPERGRAPH is not **coNP**-hard, unless any **coNP**-complete problem can be solved in subexponential time, and gives evidence that it may lie in an intermediate class between **P** and **coNP**. The algorithm of Fredman and Khachiyan can also be used as an oracle for generating all minimal transversals of a hypergraph in incremental subexponential time, the best provable upper time bound yet.

All previous discussion makes clear that the generation of minimal or maximal structures is crucial in many areas of Computer Science, while the difficulty of these problems is not well understand yet and their precise complexity is unknown. The purpose of this Thesis is to study the computational complexity of the generation problems defined above and to design efficient algorithms for solving them.

We firstly define suitable measures for measuring the efficiency of an algorithm for solving problems with large output. These measures take into account not only the size of the input but the size of the output, too. We next give formal definitions for the generation problems we are going to deal with, along with a number of cases in Computer Science where these problems emerge as a central subproblem. An extended literature in all cases is also given.

Our algorithmic approach of the TRANSVERSAL HYPERGRAPH GENERATION problem starts by presenting an efficient and practical algorithm for solving this problem. The proposed algorithm incorporates a number of ideas and improvements like the use of generalized nodes, the depth-first computation of minimal transversals and selective generation of the next minimal transversal using the notion of the appropriate node. We prove that our algorithm correctly generates all minimal transversals with no regeneration. Although no time bound is proved yet, we prove that its memory requirements are a polynomial to the size of the input hypergraph. Moreover, experimental evaluation has shown that the algorithm behaves well in practice, it produces the output with surprising uniformity, and it can successfully handle large problem instances.

For the MAXIMAL MODEL GENERATION problem, we present a resolution-like procedure that removes all positive occurrences of any variable in an arbitrary Boolean expression in CNF, and thus, transforms it into a purely negated one. We prove that this procedure preserves the set of maximal models of the initial expression and, hence, the generation of its maximal models can be done by any algorithm that solves TRANSVERSAL HYPERGRAPH GENERATION. The resulting purely negative expression may by exponentially larger that the initial one. This is not caused by the proposed procedure, since, as we show, even for simple Horn expressions the smallest purely negative expression equivalent with respect to its maximal models, is exponentially larger. We prove, however, that the generation of the maximal models of an expression with constant number of variables having positive appearance can by done in incremental subexponential time. In case that the number of variables having positive appearance in the expression is $O(\log\log m)$, where m is the number of clauses, then the generation of the maximal models can be done in incremental $O(m^{poly(\log m)})$ time. Using the resolution-like procedure, we finally present a polynomial delay algorithm (that is, an algorithm where the time between two consecutive output bits is a polynomial to the size of the input) for the 2CNF case.

We studied the computational complexity of MAXIMAL MODEL GENERATION starting from simpler, but also difficult, problems related with it. We prove that the problem of checking whether a model is maximal is **NP**-complete for general CNF expressions while it is polynomial for the polynomial cases of SATISFIABILITY. For general CNFs, the problem of checking whether there exists another maximal model, apart from some given ones, is also **NP**-complete. Obviously, for the difficult cases of SATISFIABILITY, MAXIMAL MODEL GENERATION is difficult, too. We prove that for the Horn case, the problem of checking whether there exists another maximal model, apart from some given ones, is **NP**-complete. Hence, unless $\mathbf{P} = \mathbf{NP}$, these is no output polynomial time algorithm for generating all maximal models of a Horn expression. We also show that MAXIMAL MODEL GENERATION can be efficiently solved (with polynomial delay) for the 2CNF case, while the existence of an output polynomial time algorithm for the affine case is stated as an intriguing problem.

In general, MAXIMAL MODEL GENERATION is at least as hard as TRANSVERSAL HYPERGRAPH GENERATION. The existence of an output polynomial time algorithm for TRANSVERSAL HYPERGRAPH GENERATION would result in a polynomial time algorithm for DUALIZATION; the opposite direction also holds. Based on the deterministic algorithm of Fredman and Khachiyan, we prove that DUALIZATION can be solved with limited nondeterminism: We present a nondeterministic algorithm that solves DUALIZATION in polynomial time plus $O(\log^2 n)$ nondeterministic guesses, where n is the size of the input. We, thus, place DUALIZATION in $\mathbf{coNP}[\log^2 n]$, the subclass of **coNP** where only the first $O(\log^2 n)$ steps are nondenerministic. This result implies a better upper time bound for the complexity of the problem and makes straightforward the subexponential running time of the deterministic algorithm.