Foundations of Databases and Query Languages
Exercise 6: FO Query Optimisation and Trakhtenbrot's Theorem
5 June 2015

Excercise 6.1 Use Trakhtenbrot's Theorem to show that the following problems are undecidable by reducing finite satisfiability to each of them:

(a) FO query containment

(b) FO query emptiness

(c) Domain independence of FO queries

Excercise 6.2 In the lecture, we have seen a logical formula that is finitely satisfiable if and only if the given deterministic Turing machine (DTM) halts after finitely many steps on the given input. Reconsider this formula (including corrections made after the lecture).

For each of the following statements, decide if it is true or false. Justify your answer in each case by explaining why the statement does (or does not) follow from the formula.

(a) If the formula has a model at all, then this model is finite.

(b) Every model contains a "start configuration": a right-sequence of elements ("cells") that are not reachable from any other cell via future, and where there is a first element in the chain (a cell with no element to its left).

(c) Every model contains exactly one such start configuration.

(d) If a cell is reachable from the first cell of the start configuration via future, then it does not have a cell on its left.

(e) The future of a cell's neighbour is equal to the neighbour of the cell's future.

(f) If the Turing machine halts on the input, then every model of the formula is finite.

(g) No cell can ever reach itself via future, i.e., there is no loop in the future relation.

Excercise 6.3 In the lecture, we have seen a logical formula that is finitely satisfiable if and only if the given deterministic Turing machine (DTM) halts after finitely many steps on the given input. Extend this definition so that the resulting formula is satisfiable if and only if:

(a) a given non-deterministic TM halts after fintely many steps.

(b) a given DTM halts after at most n steps (for a given number n).

(c) a given DTM halts after at most 2^n steps (for a given number n).

Make sure that your encoding is polynomial in n.

Excercise 6.4 The set of all first-order tautologies is recursively enumerable. Use Trakhtenbrot's Theorem to show that this is not the case under finite model reasoning.

Recall that a language is recursively enumerable if there exists a (possibly non-terminating) algorithm that produces any word from the language after finite time.