Astérisque

ERWIN ENGELER

Lover bounds by Galois theory

Astérisque, tome 38-39 (1976), p. 45-52

<http://www.numdam.org/item?id=AST_1976__38-39__45_0>

Société Mathématique de France
Astérisque 38-39 (1976) p.45-52

LOWER BOUNDS BY GALOIS THEORY

by

Erwin ENGELER

One of the most successful applications of classical Galois Theory has to do with the discussion of solvability of polynomial equations $p(x) = 0$: given solution algorithms for polynomial equations $q_1(a,y) = 0, \ldots, q_s(a,y) = 0$ with parameters a, say $q_r(a,y) \equiv y^r - a$, can one compose the solution algorithms for the q_r to one for p ? The method of classical Galois theory consists in comparing field extensions and corresponding groups of automorphisms and yields a necessary and sufficient condition for the relative solvability of p in terms of the solvability of the Galois groups of p (and $q_1, \ldots, q_s$). If the underlying field is provided with an effective irreducibility criterion, the the group of p can be effectively determined and a relative solution algorithm for $p(x) = 0$ (with respect e.g. to the extraction of radicals) can be effectively constructed from the knowledge fo the structure of the solvable group of p. In addition, one observes a direct relationship between the size of the group of p and the complexity of the solution algorithm.

In a previous paper [2] we have investigated the possibility of extending the above paradigm of classical Galois theory to areas of application not involving fields, making use of a suitable generalization of Galois theory developed earlier, [1]. The purpose of the present note is to summa-

rize briefly the main results and to give a very simple explicit example, whose main purpose is to illustrate some of the tools developed for applied Galois theory, mainly for the discussion of lower bound on complexity.

1.- SUMMARY OF DEFINITIONS AND RESULTS [1,2]

1.1.- Let Γ be an universal first-order theory, a an additional individual constant, L_o the set of quantifier-free formulas of the language of Γ, $L_o(a)$ its extension by a. Let $\Delta(a)$ be a diagram in $L_o(a)$ consistent with Γ and let $\varphi(a,y)$ be a formula of $L_o(a)$ with the one free variable y indicated. We call $\varphi(a,y)$ a well-posed problem if there exists a splitting diagram $\Delta(a,b_1,\ldots,b_n) \supseteq \Delta(a)$ in an extension $L_o(a,b_1,\ldots,b_n)$, i.e. a diagram satisfying :

$$\Gamma \cup \Delta(a,b_1,\ldots,b_n) \vdash \varphi(a,b_i), \quad i = 1,\ldots,n ;$$

$$b_i \neq b_j \in \Delta(a,b_1,\ldots,b_n) \quad \text{iff } i \neq j ;$$

$$\Gamma \cup \Delta(a,b_1,\ldots,b_n) \cup \varphi(a,b) \vdash \bigvee_{i=1}^{n} b = b_i .$$

1.2.- If Γ has the amalgamation property (i.e. if for any models $\underline{A}_1$, $\underline{B}_1$, $\underline{B}_2$ and monomorphisms $\underline{A} \to \underline{B}_1$ and $\underline{A} \to \underline{B}_2$ there exists a model $\underline{C}$ and monomorphisms $\underline{B}_1 \to \underline{C}$ and $\underline{B}_2 \to \underline{C}$ such that the square commutes), then : if $\varphi(a,y)$ is well-posed with respect to $\Delta(a)$ and $\Delta(a,b_i,\ldots,b_n)$ and $\Delta'(a,b_i,\ldots,b_n)$ are splitting diagrams, then there exists a permutation $s \in S_n$ such that $\Delta'(a,b_1,\ldots,b_n) = \Delta^s(a,b_1,\ldots,b_n)$, the result of substituting $b_{s(i)}$ for b_i in all formulas of $\Delta(a,b_1,\ldots,b_n)$.

1.3.- The group of the problem $\varphi(a,y)$ is defined by $G^*_{\Delta(a)}(\varphi) := \{s \in S_n : \Gamma \cup \Delta(a,b_1,\ldots,b_n) \vdash \rho \equiv \rho^s \text{ for all } \rho \in L^*_o\}$ where L^*_o is a sublanguage of $L_o(a,b_1,\ldots,b_n)$ containing the vocabulary of φ. If Γ has the amalgamation property with respect to L^*_o and φ is well-posed, then $G^*_{\Delta(a)}(\varphi)$ depends only on $\Delta(a)$ and is a group. Indeed it is isomorphic to the

group of all automorphisms of the minimal model $\underline{A}(a,b_1,\ldots,b_n)$ of $\Gamma \cup \Delta(a,b_1,\ldots,b_n)$ leaving the minimal model $\underline{A}(a)$ of $\Gamma \cup \Delta(a)$ pointwise fixed.

1.4.- Assume that $\varphi(y)$ <u>is solvable relative to problems</u> $\psi_i(u,v)$, for $i = 1,\ldots,s$, by an algorithm π in all models of $\Gamma \cup \Delta$, where Δ is a diagram for L_o consistent with Γ.

To simplify notation, choose $s = 1$. Let w be a finite terminating path through π consistent with $\Gamma \cup \Delta$. Let $\underline{A}$ be the minimal model of $\Gamma \cup \Delta$. Along w we encounter a finite number of calls of the auxillary algorithm for the solution of problems $\psi(\tau,v)$, where each time τ is a term built up from constants, operations and solutions of earlier calls of the auxillary algorithm. Thus, path w corresponds to a sequence of extensions of $\underline{A}$ as follows :

$$\underline{A} \xrightarrow[\psi(\tau_1,v)]{} \underline{A}(a_{1,1},\ldots,a_{1,m_1}) \longrightarrow \cdots$$

$$\cdots \longrightarrow A(a_{1,1}\ldots,a_{q,m_q}) \xrightarrow[\psi(\tau_{q+1},v)]{} A(a_{1,1},\ldots,a_{q+1,m_{q+1}}) \longrightarrow \cdots$$

$$\cdots \longrightarrow A(a_{1,1},\ldots,a_{k,m_k}) \ ;$$

where $\tau_1 \in L_o$, $\tau_{q+1} \in L(a_{1,1},\ldots,a_{q,m_q})$. The solutions are terms in $L_o(a_{1,1},\ldots,a_{k,m_k})$, say $b_1 := \sigma_1,\ldots, b_n := \sigma_n$. The corresponding sequence of groups is defined by :

$$G_q^* := \{t \in P : \Delta^* t(a_{1,1},\ldots,a_{k,m_k}) = \Delta^*(a_{1,1},\ldots,a_{k,m_k}),$$

$$t(a_{i,j}) = a_{i,j} \text{ for all } i \leqslant q\} \ ;$$

P is the set of all permutations of $\{a_{1,1},\ldots,a_{k,m_k}\}$ which respect the sets $\{a_{q,1},\ldots,a_{q,m_q}\}$, $q = 1,\ldots,k$.

1.5.- Main results :

1.5.1. $G^*_\Delta(\varphi) \cong \{s \in S_n : \exists\, t \in G^*_o \text{ s.t. } \Gamma \cup \Delta(a_{1,1},\ldots,a_{k,m_k}) \vdash \sigma_{s(i)} = \sigma^t_i \quad \forall i\}$;

1.5.2. $G^*_\Delta(\varphi) \lhd G^*_o$, $G^*_{q+1} \lhd G^*_q$, $q = 0,\ldots,k-1$;

1.5.3 $G^*_q/G^*_{q+1} \cong G^*_{\Delta(a_{1,1},\ldots,a_{q,m_q})}(\psi(\tau_{q+1},v))$, $q = 0,\ldots,k-1$;

1.5.4. The number of calls of an algorithm for $\psi(\tau,v)$ in obtaining all solutions of φ is bounded below by

$$\ell n\, |G^*_\Delta(\varphi)| / \ell n \max \{|G^*_{\Delta(a)}(\psi(a,v))| : \Delta \subseteq \Gamma \cup \Delta(a) \text{ cons.}\}.$$

2.- THE GROUP OF KNAPSACK PROBLEM

Let us consider a knapsack problem posed by giving positive natural numbers $a_1,\ldots,a_n$ and b and asking for numbers x_i, $i = 1,\ldots,n$, $x_i \in \{0,1\}$, for which :

$$\sum_{i=1}^{n} a_i x_i \leqslant b.$$

We imagine an algorithm which produces the desired x_i, given inputs $a_1,\ldots,a_n$ and b. To fix ideas, such an algorithm is assumed to build up a solution $< x_1,\ldots,x_n >$ step by step from the empty sequence $\emptyset$ through sequences $< x_1,\ldots,x_i >$ to the final solutions. Thus, the domain on which the theory operates consists on the one hand of the natural numbers (with their usual arithmetic) and the set of sequence of 0's and 1's of length at most n. Since, reasonably, $b \leqslant \Sigma\, a_i$, we may restrict the arithmetic to $\{0,1,2,\ldots,\sum_{i=1}^{n} a_i\} = D \subseteq N$.- To apply our general theory, we now have to set up a formal language in which to axiomatize the relevant aspects of this two-sorted domain, formulate problems and auxillary problems, establish the hypotheses of the theory and produce the groups.

2.1.- As a language we choose a two-sorted first-order language with :

variables $x,y,z,\ldots$ for states (i.e. finite sequences of 0's and 1's),

i,j,k for numbers in the finite set D ;

constants $\emptyset$ for the (empty) starting state,

$a_1,\ldots,a_n,b$ the parameters of the problem ;

function symbols $+,.$ for the (partial) arithmetic operations in D,

p_i the projection functions $p_i(x) = x_i \in \{0,1\}$,

ℓ the length function $\ell(<x_1,\ldots,x_s>) = S$;

predicates $\leqslant$, $=$ on numbers in D.

This basic language is extended by function symbols and predicate symbols, whose meaning is defined below. The new symbols will serve to constitute the vocabulary of the group language L_o^*.

$$B(x) :\equiv \sum_{i=1}^{\ell(x)} a_i x_i \leqslant b ;$$

$$C_k(x) :\equiv \sum_{i=1}^{\ell(x)} a_i x_i = k \wedge k + \sum_{i=\ell(x)+1}^{n} a_i > b,\ k = 0,1,\ldots,\Sigma a_i ;$$

$$x S y :\equiv y = <x_1,\ldots,x_s,x_{s+1}> \wedge x = <x_1,\ldots,x_s> \text{ for some } s ;$$

$$x = P(y) :\equiv x S y \vee (x = \emptyset \wedge y = \emptyset).$$

2.2.- The <u>axiomatization</u> Γ formulates the obvious true sentences about the arithmetic in D and the basic properties of the set of states relative to the projection functions, length, B,P,S etc... The minimal model of Γ has just one state, $\emptyset$, its diagram Δ is completely determined by the numerical values of the parameters $a_1 \ldots,a_n$ and b.

2.3.- The <u>knapsack problem</u> as well as the <u>auxillary problems</u> have now to be formulated in the group language L_o^*, in our case as quantifier-free formulas over B,S and P.

$$\text{knapsack}(x) :\equiv B(x) \wedge P^n(x) = \emptyset \wedge P^{n-1}(x) \neq \emptyset ;$$

$$\text{expand}(x,y) :\equiv x S y \wedge B(y).$$

In addition, we need to convince ourselves, that the set of solutions of the knapsack problem can be obtained by repeatedely solving the auxillary "expand" problem. This can be done by an obvious loop-free program, sketched below.

```
begin  x : = ∅ ;
       x0 , x1 : = expand (x,.) ;
       x00 , x01 : = expand (x0,.) ;
       ...
       if (x00..0 is defined) then a_1 : = x00..0 else...
       ...
end.
```

We are making use of the fact that the knapsack problem and the expand problems are well-posed ; indeed, the set of possible states is finite, there are at most two states which expand a given state and satisfy B, and any sets of states which are full solution sets for the knapsack problem are identical (up to enumeration) subsets of the set of all possible states.

2.4.- The predicate symbols $C_k(.)$, which are not used for the formulation of the problems, are put into the group language in order to guarantee the amalgamation property. The state part of any model of Γ may be visualized as an initial section of the full binary tree of depth n. To prove the amalgamation property it suffices to show how an isomorphism f (with respect to B, C_k, S, P) of two initial subtrees can be expanded to an additional node. Consider, therefore, a branch x which is to be prolonged.

First case : $C_k(x)$ holds for some k. Then x remains pointwise fixed under f, since for each node on x there is a C_j which is valid for it, and the sequence of valid C_js determines the branch x completely. The two possible expansions x' and x" of x are distinguishable in the group language because at least one of them satisfies a C_j, but not both the same. Thus, f may be expanded to f' by $f'(x') = x'$, $f'(x") = x"$.

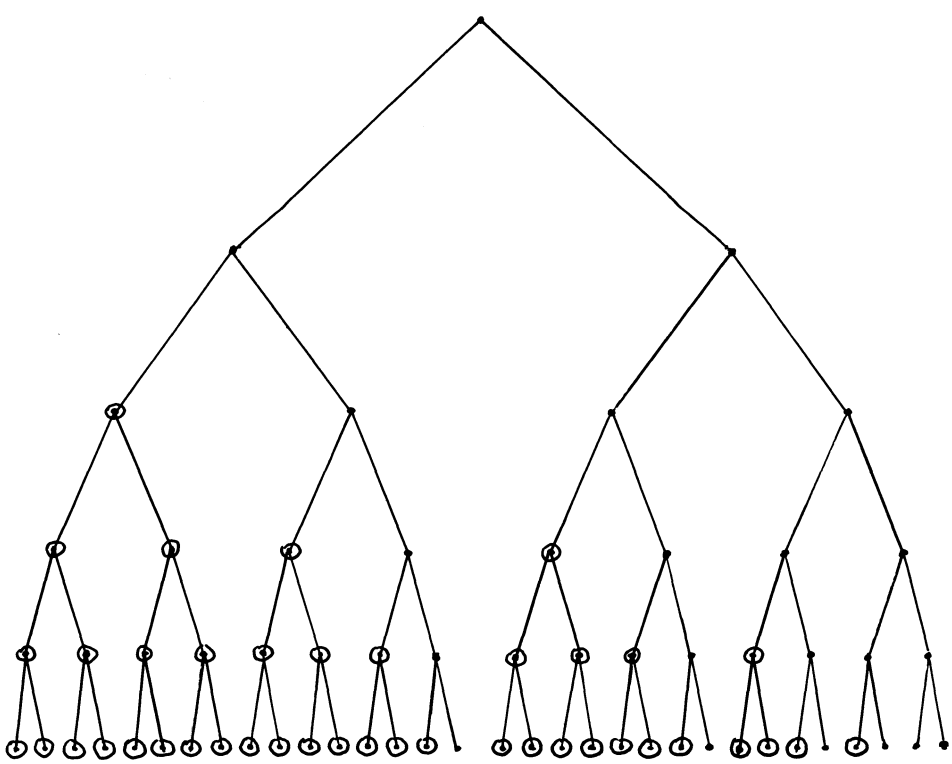
Second case : $C_k(x)$ holds for no k. Then no C_k holds for x', the expansion of x. Hence f can be expanded to x' in the obvious ways.

2.5.- The group G_Δ^* is determined (as to its size) for the problem $2x_1 + 2x_2 + 2x_3 + 2x_4 + 2x_5 \leqslant 7$ by considering isomorphism with respect to B, C_k, S, P of 26 solutions represented by the circled final nodes of the tree below. The other circled nodes are those for which no C_k holds.

An easy computation yields :

$$|G_\Delta^*| = 2^7 \cdot 2^3 \cdot 2^1 \cdot 2^0 \cdot 2^3 \cdot 2^1 \cdot 2^0 \cdot 2^1 \cdot 2^0 \cdot 2^0 = 2^{16} .$$

Since the size of the group of the expand problem is at most 2, our general theorem yields as a lower bound of complexity $\log_2 2^{16} = 16$.



REFERENCES

[1] ENGELER E. : On the Solvability of Algorithmic Problems, in : H.E Rose and J.C. Shepherdson (eds) Logic Colloquium 73, North-Holland Publ. Co, 1975, pp. 231-251.

[2] ENGELER E. : Structural Relations between Programs and Problems, to appear as invited talk in : J. Hintikka (ed.) Proceedings of the 5th Intern. Congress of Logic, Methodology and Philosophy of Science. Preprint : Report N°15 of the Institut für Informatik, E.T.H., Oct. 1975.

Erwin ENGELER
Institut für Informatik
Clausiusstrasse 55
CH - 8006 ZÜRICH (Suisse)