Computation of Galois Group Elements of a Polynomial Equation

By E. J. Cockayne*

Abstract. This note demonstrates the use of the computer for constructing elements of the Galois group over the rationals of a polynomial equation with rational coefficients. ■

1. The Principal Theorem Involved. Any polynomial equation in x' with rational coefficients can be transformed by $x' = \lambda x$ (for some rational λ) into a polynomial equation in x which has integer coefficients and is monic. Such a transformation preserves Galois groups over the rationals and it is therefore sufficient to consider polynomial equations of this simpler type.

The methods of this paper depend on the following theorem [1, pp. 190–191]:

Let p be any prime number, $I/(p)$, the residue class ring of integers modulo p and R the field of rationals. Suppose that $f(x)$ reduces to $f_p(x)$ modulo p, neither $f(x)$ nor $f_p(x)$ has a multiple root and $f_p(x)$ has the irreducible factorisation

$$f_p(x) = f_1(x)f_2(x)\cdots f_r(x)$$

in $I/(p)$ where these factors have degrees $d_1, d_2, \cdots, d_r$ respectively. Then G, the Galois group of $f(x) = 0$ over R, contains a permutation whose representation as a product of disjoint cycles consists of r cycles of lengths $d_1, \cdots, d_r$.

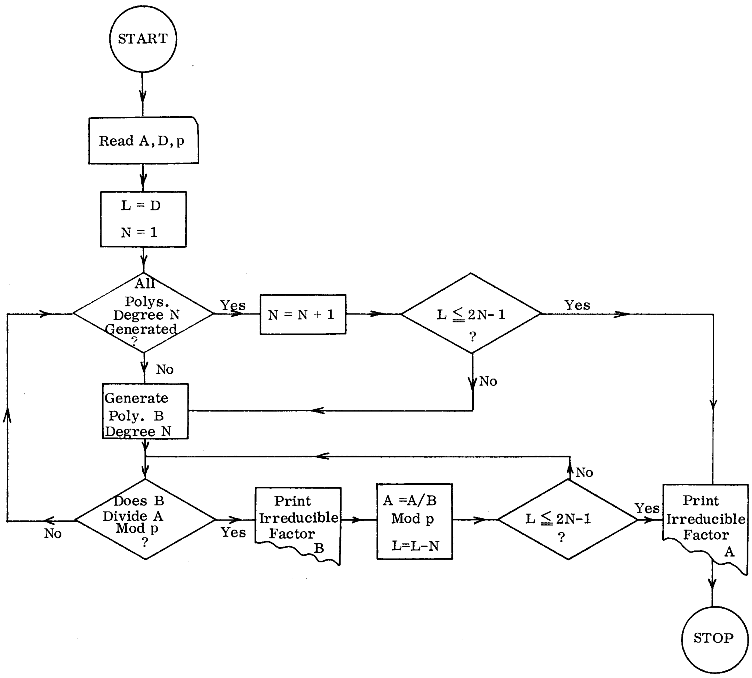
2. Outline of Procedure. For a series of primes p, the irreducible factors of $f(x)$ modulo p are calculated on the machine (see Section 3) and printed out together with their degrees $(d_1, \cdots, d_r)_p$. The polynomial $f(x)$ and its reduced polynomial modulo p are tested for multiple roots by inspection of the factors and using the results of [1, p. 120].

3. Construction of Irreducible Factors Modulo p of an Integer Polynomial. The procedure given in the flow chart determines the irreducible factors modulo p of the polynomial degree d whose coefficients are initially stored in vector A. At any stage, L is the degree of the polynomial stored in A. The algorithm generates successively all monic polynomials B over $I/(p)$ of degree $N = 1, 2, 3, \cdots$ in this ascending order. As each B is generated, we determine by standard polynomial division whether or not it is a factor of A modulo p. Any factor B thus found is certainly irreducible, for any factors of B would have been noticed at a smaller value of N. The process is continued by replacing A and L respectively by the quotient A/B mod p and its degree, and by testing the new dividend A with the same divisor B.

Received October 3, 1968.

* Visiting Lecturer at the University of Auckland, Auckland, New Zealand, during academic year 1968–69.

The algorithm terminates when $L \leqq 2N - 1$ and the current value of A is reduced modulo p and printed out as the last irreducible factor. For suppose the contrary: A degree L has a factor mod p of degree N where $L \leqq 2N - 1$. Then A also has a factor of degree $L - N \leqq N - 1$ which would have been extracted at an earlier stage.



Procedure for Irreducible Factors modulo p of an Integer Polynomial

The FORTRAN program, to construct irreducible factors modulo p, is reproduced in the microfiche section of this issue.

4. Galois Group Properties from the Algorithm. The algorithm produces a set P of primes and for each $p \in P$ a set of integers $\{d_1, \cdots, d_r\}_p$. Assuming no trouble with multiple roots, for each $p \in P$, G contains a permutation α_p as described in Section 1 and hence S_p, the cyclic permutation group generated by α_p is a subgroup of G with order the least common multiple of $\{d_1, \cdots, d_r\}_p$. Thus we obtain information about the order of G. The disjoint cycle structure of any element of S_p may be calculated using the following result: If β is a cycle of length n, then in disjoint cycles β^t contains exactly d cycles of length n/d where $d =$ g.c.d. (n, t). Finally, if our methods produce a transposition and an $(n - 1)$ cycle as elements of G for a polynomial of degree n where G is known to be transitive (this is true if $f(x)$ is irreducible modulo any prime), then $G = S_n$, the symmetric group of all permutations of n objects.

5. An Application. In [2] Z. A. Melzak showed that the classical Steiner problem, to join n points in the Euclidean plane by a minimum length network, could be solved by a finite number of Euclidean constructions (i.e. ruler-compass constructions in the classical sense). The problem is also generalized so that more complicated network functions than length are to be minimized. $S_{n\alpha\beta\gamma}$: Given nonnegative reals α, β, γ and n points a_i ($i = 1, \cdots, n$) in the plane to find an integer k ($\geqq 0$) and k additional points $s_1, \cdots, s_k$ and to construct the tree U (circuit-free connected graph) with vertices $a_1, \cdots, a_n, s_1, \cdots, s_k$ so as to minimize the sum

$$L(U) + \alpha \sum_{i=1}^{n} w(a_i) + \beta \sum_{j=1}^{k} w(s_j) + \gamma k ,$$

where $L(U)$ is the total length of the network and $w(b)$ is the valency of vertex b.

The methods of this paper were used to prove that the more general problem is not, in general, solvable by Euclidean constructions. For suitable α, β, γ, $S_{n\alpha\beta\gamma}$ reduces to (see [2]): Given n points a_i ($i = 1, \cdots, n$) in the plane to find the point q which minimizes $\sum_1^n |qa_i|$.

Five points with integer coordinates were taken, symmetrically placed with respect to the x-axis. It was shown that the x coordinate of q satisfied an irreducible eighth degree polynomial equation whose Galois group over R had odd order. Thus this coordinate was not an element of an extension field of R of degree 2^m, hence q could not be found by Euclidean constructions [1, p. 185].

6. Examples. The table lists the coefficients of polynomials $f(x)$ in descending order together with the degrees of their irreducible factors modulo 2, 3, 5, 7, 11 (unless there is a multiple root). The structure column gives cycle lengths of elements of G and N (the least common multiple of the degrees of factors) is a divisor of the order of G. For example the Galois group of

$$x^5 + 2x^4 + 8x^3 + 3x^2 + 5x + 1 = 0$$

contains cycles of length 2, 3 and 5 and two permutations whose disjoint cycle representation consist of two 2-cycles and a 2-cycle and 3-cycle respectively. The order of G is a multiple of 30.

$f(x)$	2	3	5	7	11	*Structure*	N
1 4 5 8	Multiple Root	Multiple Root	3 Irreducible	2, 1	3 Irreducible	2, 3 $G = S_3$	6
1 6 7 4 2	Multiple Root	1, 3	Multiple Root	1, 1, 2	1, 1, 2	2, 3	6
1 2 8 3 5 1	Multiple Root	5 Irreducible	1, 2, 2	2, 3	2, 3	2, 3, 2–2, 2–3, 5 Transitive	30
1 1 1 1 7 5 2	Multiple Root	1, 2, 3	Multiple Root	2, 4	6 Irreducible	2, 3, 4, 6, 2–3, 2–4	24
1 2 2 3 9 8 5 4	1, 1, 5	Multiple Root	1, 6	1, 2, 4	Multiple Root	4, 5, 6, 2–4	120

7. Acknowledgement. This work was completed while the author was a Fellow of the 1968 Summer Research Institute of the Canadian Mathematical Congress at the University of British Columbia and was partially supported by Canadian National Research Council Grant NRC A4810.

University of Victoria
B. C., Canada

1. B. L. van der Waerden, *Modern Algebra*. Vol. 1, Springer, Berlin, 1937; English transl., Ungar, New York, 1949. MR **2**, 120; MR **10**, 587.
2. Z. A. Melzak, "On the problem of Steiner," *Canad. Math. Bull.*, v. 4, 1961, pp. 143–148. MR **23** #A2767.

```
C
   48 N=N+1
  115 IF (L-(2*N-1)) 62,62,88
   62 WRITE (3,127) L,A
      CALL EXIT
   30 WRITE (3,127) L,IQUO
      CALL EXIT
  127 FORMAT ('0',I2,10X,10(I2,2X))
   38 FORMAT (10I2,2X,I2,2X,I2)
      END
```

```
      SUBROUTINE IPDIV(IA,IDIMA,IB,IDIMB,IQUO,IREM)
C
C DIVIDES INTEGER POLY IA BY INTEGER POLY IB.QUOTIENT IS IQUO
C REMAINDER IS IREM.
C
      IF (IDIMA.LT.IDIMB) CALL EXIT
      DIMENSION IA(IDIMA),IB(IDIMB),IREM(IDIMB),IQUO(IDIMA)
      K=IDIMA
    5 IF (K.GE.IDIMB) GO TO 4
      DO 8 I4=1,K
    8 IREM(I4)=IA(I4)
      RETURN
    4 IQUO(K+1-IDIMB)=IA(K)
      DO 33 J=1,IDIMB
   33 IA(K-IDIMB+J)=IA(K-IDIMB+J)-IA(K)*IB(J)
      K=K-1
      GO TO 5
      END
```

```
      SUBROUTINE VECTO(IVEC,I14)
C
C ASSIGNS 0 TO ALL COMPS OF A VECTOR.
C
      DIMENSION IVEC(I14)
      DO 99 I15=1,I14
   99 IVEC(I15)=0
      RETURN
      END
```

COMPUTER USE IN CONTINUED FRACTION EXPANSIONS

EVELYN FRANK

FORTRAN PROGRAM

```
C       FORTRAN PROGRAM
        DIMENSION A(100),C(100),B(100),D(100),P(100),R(100),Q(100)
        DIMENSION S(100),FN(100)
        WRITE(6,6000)
   6000 FORMAT('1',20X'CONTINUED FRACTION EXPANSION FOR BINOMIAL QUADRATIC
       1 SURD'/)
C       INITIALIZE INDICES AND VARIABLES
        NP=0
     98 N=0
        K=1
        S(1)=1.
        R(1)=1.
        A(1)=1.
        C(1)=1.
C       READ STARTING CONDITIONS
        READ (5,5001) P(1),Q(1),DD,IPER
        IF(Q(1).EQ.0.) GO TO 99
C       IPER = NBR OF A'S TO BE READ IN
        IPER=IPER+1
        READ (5,5002)(A(I),I=2,IPER),(C(I),I=2,IPER)
   5002 FORMAT(8F10.1)
   5001 FORMAT(3F10.1,I3)
        DX=SQRT(DD)
        FN(1)=(P(1)+DX)*(S(1)/Q(1))
        NP=NP+1
        WRITE( 6,6001)NP,P(K),Q(K),DD
   6001 FORMAT(21X'PROBLEM 'I2,7X'PO ='F7.1,4X'QO ='F7.1,4X'D ='F8.1)
        WRITE( 6,6002)
   6002 FORMAT(24X'N'3X'A'5X'C'9X'B'8X'D'8X'P'8X'R'8X'Q'8X'S'6X'FN')
        IC=1
        IP=2
        GO TO 25
     12 N=N+1
        IF(K.EQ.98) GO TO 120
        K=K+1
        A(K)=A(IC)
        C(K)=C(IC)
        PP=B(N)*Q(N)*R(N)-D(N)*S(N)*P(N)
        RR=D(N)*R(N)*S(N)
        CALL LTU(PP,RR)
        P(K)=PP
        R(K)=RR
        QQ=(DD*RR*RR-PP*PP)*C(IC)*S(N)
        SS=A(IC)*Q(N)*RR*RR
        CALL LTU(QQ,SS)
        Q(K)=QQ
        S(K)=SS
        FN(K)=(PP+DX)*(SS/QQ)
C
C        'B' ROUTINE
C
     25 BINC=S(K)
C       B(K) IS A MULTIPLE OF S(K)
C       BINC IS THE INCREMENT
```