FAST IDENTIFICATION OF PRIMITIVE POLYNOMIALS OVER GALOIS FIELDS: RESULTS FROM A COURSE PROJECT

Klaus Krogsgaard[†]

Texas Instruments Inc.
12500 TI Boulevard, MS 8708
Dallas, TX 75243
k-krogsgaard@ti.com

Tanja Karp

Texas Tech University
ECE Dept., Box 43102
Lubbock, TX 79409-3102
tanja.karp@ttu.edu

ABSTRACT

In this paper, we present the format of a graduate course in digital communications fostering course projects, active student participation, and communication among students. We illustrate how electrical engineering students show increased interest in theoretical mathematical concepts, if motivated by a design problem, and are actually able to improve the performance of a state-of-the-art software package.

1. INTRODUCTION

During their first industry appointment, electrical engineers generally face the experience of a large gap between concepts learned in class and the complexity of the system design, they get involved in. In addition to understanding the overall system, their part in it, and its interfaces, they simultaneously struggle with new software packages, communicating and collaborating with new colleagues, and catching up with all the special terms and abbreviations used in the industrial environment.

Advanced graduate courses in digital signal processing and communications try to close this gap by offering topics such as wavelet based image compression, spread spectrum communications, multicarrier modulation, and audio compression to the curriculum. Traditional teaching methods using blackboard, slides, and software presentations allow the instructor to explain these complicated systems and designs in form of block diagrams followed by a detailed discussion of each building block, and the demonstration of the algorithm using a software package. The students, however, are placed in the passive role of digesting the given information and memorizing it for in-class tests. Neither their problem solving skills nor their communication skills are challenged. Many of them do not really understand the design details and remember the course as one "that involved a lot of nasty math".

[†] This work was performed as a graduate student participating in the Dual Master's Degree Program between Texas Tech University and the Technical University of Denmark.

In this paper we present a project oriented approach to teach such courses and show how the outcome of such a student project can actually impact the performance of a state-of-the-art software package.

2. PROJECT ORIENTED COURSE WORK

In this section, we describe a teaching format aiming at having students experience a more engineering like environment and boosting their creativity, enthusiasm, team work, and problem solving skills.

2.1. Course Format

The three credit hour course on "Topics in Advanced Communications" offered as part of the Electrical Engineering Graduate Program at Texas Tech University was split into three terms of one month each. During each term, a different communications system was studied. At the first class meeting, the instructor, who is the second author of this paper, presented a list of possible topics such as CDMA, blind equalization, OFDM, DMT, etc., and students were adding their ideas. Each student then casted votes for his/her favorite three topics and the three most popular ones were chosen as course topics. Also, the class discussed different options to implement the final system in software or hardware and decided that MATLAB [1] with its toolboxes provided a good tool, since it is a state-of-the-art software package that is widespread in industry and allows fast system design thanks to its built-in functions. It was interesting to note that many new international graduate students were not familiar with MATLAB and had never used this tool before.

A general system introduction for each topic was given by the instructor. Next, the system was broken down into subsystems with well defined interfaces suitable for groups of 2-3 students to implement. This approach enables the students as a class to cover larger and more complex systems and projects, while at the same time allowing students to diversify and form groups based on individual interests.

Since the project topics were new to the students, it enabled them to look at the task at hand with new eyes. At the end of the term, the groups had to join their subsystems to obtain a functioning system.

To keep students interested in other groups designs, partial project credit was given for the functionality of the overall system. Also, students knew that the test at the end of the term would contain questions regarding all subsystems and designs.

Students were asked to team up with different classmates for each project. This forces students of all nationalities to work together and communicate with each other. In addition, it facilitates a more fair grading procedure since weak or lazy students cannot hide behind the work of the same strong students for all projects. At the same time, this procedure simulates a more realistic working environment where you cannot choose your colleagues. Each topic was graded based on the students' design, the quality of the presentation they gave in order to explain their subsystem to their classmates, and the results of a take-home test and an in-class test, evaluating individual student's problem solving skills and system understanding.

2.2. Classroom Setting

Class meetings were used to gather and discuss information sources found by students during their research of the topic, make these sources available to all groups at the teaching website, discuss the groups' progresses or difficulties, clarify interfaces, and explain theoretical aspects of possible algorithms, their possible performance, computational cost, and real-time requirements. Students were experiencing a classroom environment, where they could openly ask questions, were expected to participate in problem solving discussions, and had to explain their designs to classmates. The success of the overall project depended on all students.

Students who were involved in the design of a subsystem that was discussed in a class meeting had to explain its purpose, design aspects, and algorithms. Since the rest of the students had to understand the subsystem in order to pass the test, they were more alert and critical than in normal student project presentations, where the audience in general shows only a moderate interest in the outcome of the other projects. Presenting students experienced the challenge of clearly explaining a subsystem to their classmates.

The role of the instructor was mainly to direct discussions, ask critical design questions not thought of by the students, ensure understanding of the topic for all groups, help students with problems or verify proposed solutions, and clarify students presentations where they were unbalanced or wrong. In addition, the instructor explained the core subsystems and algorithms of the system with the necessary theoretical depth.

2.3. Outcome

While traditional teaching requires the instructor to prepare the course material and think about its presentation form in advance, the described course format included a large teaching-on-demand component. While having a broad idea about the topics that should be covered during certain class meetings, the instructor never knew what questions would arise in relation to the design projects. It thus required a profound knowledge of the topic from the instructor. Nevertheless, it was inevitable, that the instructor sometimes was unable to answer a question. The approach taken in this course was to take the question as a homework assignment for the class *and* the instructor in order to solve it for the next class meeting. Students thus experienced that they had acquired a level of knowledge and critical thinking, that enabled them to ask nontrivial questions, which increased their self-esteem and prepares them for creative research. Also, since the answer to their questions was often crucial to their project design, they experienced the difficulty that sometimes arises in solving a problem statement. As opposed to a problem statement in a homework assignment, there was not the alternative to simply skip it.

The experiences from the course are, that students find the use of real-life case studies and applications for in-class theory more inspiring and rewarding than through synthetic problems and projects. Compared to a stand-alone pure laboratory course, the chosen format ensures synchronization between theoretical background material and practical design. Students actually appreciated the concepts and theory that were taught since it helped them in improving their design. Facing new problem statements and trying to solve them with already acquired knowledge can often lead to alternative solutions, which are radically different than what is commonly used, and in some cases innovations and/or improvements over existing performance, as will be shown in the remaining part of this paper. By the end of the term, all students evaluated the quality of the course with 5 out of 5 possible points.

3. PRIMITIVE POLYNOMIALS

A polynomial $p(x)$ of order n over a Galois Field $\mathrm{GF}(q)$ is defined as [3]

$$p(x) = x^n + \alpha_{n-1}x^{n-1} + \ldots + \alpha_1 x + \alpha_0, \qquad (1)$$

where the coefficients α_i are members of $\mathrm{GF}(q)$, i.e. integers ranging from 0 to $q-1$. The polynomial is called irreducible in $\mathrm{GF}(q)$ if $p(x)$ cannot be factored into a product of lower-degree polynomials. An irreducible polynomial $p(x) \in \mathrm{GF}(q)$ of degree n is said to be primitive if the smallest positive integer l for which $p(x)$ divides $c(x) = x^l - 1$ is $l = q^n - 1$.

The use of primitive polynomials over Galois Fields is widely spread in digital communications applications rang-

ing from generation of pseudo-noise sequences, generator polynomials for forward error-correction block codes and convolutional codes, to spreading functions for CDMA systems. The polynomial can be easily implemented as weights for an autonomous linear feedback shift register (ALFSR), creating the pseudo-random sequence at the shift register output. When discussing CDMA as a topic of the course, students were therefore asked to prove that the short code used in CDMA [2] actually is a primitive polynomial over GF(2).

3.1. Standard Identification Procedure

The definition of primitive polynomials over $\mathrm{GF}(q)$ is easily converted into a stepwise procedure for the polynomial $p(x)$ with order n:

1. Is the polynomial $p(x)$ irreducible?

   Check that it cannot be divided by any polynomial over $\mathrm{GF}(q)$ of order $1 < m \leq \lfloor\sqrt{n}\rfloor$ without remainder

2. Is the irreducible polynomial primitive?

   Divide $c(x) = x^l - 1$ by $p(x)$ for all $l \in \mathbf{N} < q^n - 1$, and check that all divisions have remainders.

Step 1 is relative basic. In fact, it is sufficient to check only primitive polynomials of lower order, if available, or one can apply Berlekamp's reducibility criterion [5].

Step 2, however, is more complex. The straight forward approach for solving step 2 is a direct implementation of the definition by constructing test polynomials $c(x) = x^l - 1$ for increasingly higher orders l and determining the remainder $r_l(x) = c(x) \mod p(x)$. This is done for each iteration from $l = n$ to $l = q^n - 2$, unless $r_l(x) = 0$ is encountered. A break is triggered in the event of an even division, and the polynomial is classified as non-primitive. The core loop is shown in MATLAB notation below, where `gfdeconv` is a function of MATLAB's Communications Toolbox and performs the polynomial division (note that `gfdeconv` expects polynomial coefficients to be entered into vectors in ascending order).

```
for l = n:(q^n-2)
   c = [q-1 zeros(1,l-1) 1];
   [quotient,remainder] ...
      = gfdeconv(c,p,q);
   if remainder == 0
      % polynomial not primitive
      break;
   end;
end;
```

For the CDMA example, the polynomial over $\mathrm{GF}(2)$ generating the short code is of order 15. All students started off by implementing the above procedure and then ran the simulation. In doing so, they quickly found that the computational complexity was too high to obtain results within a reasonable time frame.

The computational cost of determining the remainder of $c(x)/p(x)$ is $\mathcal{O}(l)$, i.e. it increases linearly with the degree l of $c(x)$. Using

$$\sum_{l=n}^{q^n-1} l = \frac{q^n(q^n-1)}{2} - \frac{n(n-1)}{2} = \frac{q^{2n} - q^n - n^2 + n}{2}$$

to account for the `for`-loop we obtain the complexity of the implementation as $\mathcal{O}(q^{2n})$. For the short-code polynomial over $\mathrm{GF}(2)$ in CDMA the overall complexity is thus given by $\mathcal{O}(2^{30}) \approx 10^9$.

Realizing that this implementation is computationally too expensive, students then looked at functions available in the MATLAB Communications Toolbox and realized that `gfprimck(p,q)` performs the required task for them. However, the way `gfprimck` is implemented in MATLAB's Communications Toolbox 2.1 [1] is exactly the approach described above, thus using this function did not result in any speed increase. Subsequently, most students decided to simply scale down the problem statement to a toy example, where the polynomial $p(x)$ was a primitive polynomial of order 3 and the students were able to show the correct functionality of their MATLAB function within feasible time. Others, however, were challenged by the problem statement and thought of different ways to speed up the procedure. The first author of this paper finally came up with a solution that is of order $\mathcal{O}(q^n)$, which will be presented next.

3.2. Optimized Procedure

Reformulating step 2 of the primitive polynomial definition enables a more efficient use of intermediate results from the previous iterations and reduces the computational cost to one division by $p(x)$ per iteration.

The modification proposed is to replace step 2 with a check whether $x^l/p(x)$ results in a remainder $r_l(x) \neq 1$ for all $n \leq l < q^n - 1$ and in a remainder of 1 for $l = q^n - 1$. Using modulus arithmetic, one can easily verify that this is identical to the original expression. The resulting procedure now becomes very similar to performing polynomial division by hand, starting out with only the highest order coefficients of the numerator. The remainder is then combined with the next lower order coefficient, and the next iteration can start. Figure 1 shows this for a third order polynomial $p(x) = x^3 + x + 1$ in $\mathrm{GF}(2)$ and $l = 4$ through 6.

If the remainder $r_l(x)$ at iteration l is known, then the remainder for the next iteration can be obtained as $(x \cdot r_l(x)) \mod p(x)$ instead of $x^{l+1} \mod p(x)$. The corresponding MATLAB code for this new procedure is given below.

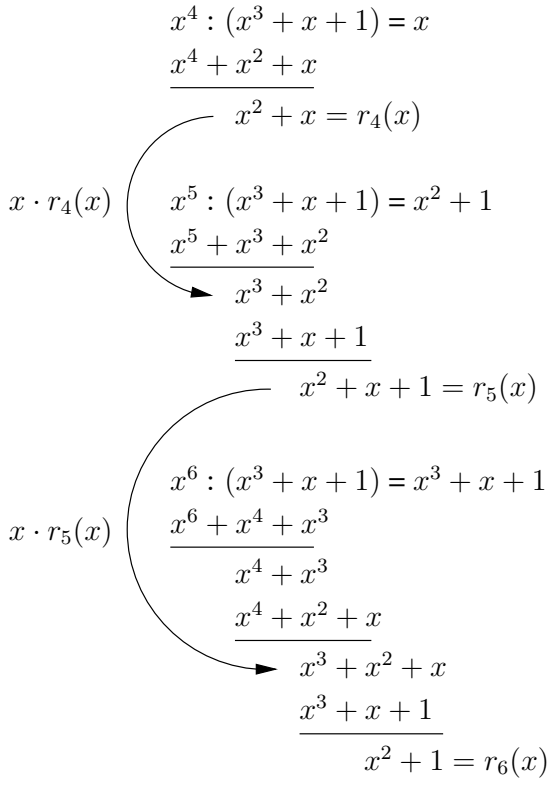

Fig. 1. Polynomial division

```
c = [zeros(1,n) 1];
for l=n:(q^n-1)
   [quotient,remainder] ...
      = gfdeconv(c,p,q);
   if remainder == 1
      % polynomial not primitive
      break;
   else
      remainder(n+1) = 0;
      c = [0 remainder(1:n)];
   end;
end;
```

Note that although the polynomial division is once again inside a `for`-loop, this procedure reduces the degree of the numerator polynomial used for all iterations to a maximum of n. It thus keeps the computation per iteration constant to one polynomial division step. It results in a total computational cost of $\mathcal{O}(q^n)$, i.e. $\mathcal{O}(2^{15}) = 32768$ for the assigned project.

The above computational complexity estimates are based solely on the expected number of computations and do not take memory allocation and use into consideration. The optimized procedure is also more efficient in this regard, as it does not need large arrays associated with high order polynomial divisions.

4. IMPACT

Impressed by the reduction in computational complexity obtained through the new method, the authors performed two steps. Since none of them is a specialist in Galois field arithmetic, they further researched the topic and found that their approach was not new but was already stated in [3]. They also learned about further methods to reduce the overall computational cost from [4, 5]. In addition they contacted The MathWorks to inform them about the improvements they have made compared to the `gfprimck` function in the MATLAB Communications Toolbox. The MathWorks performed extensive benchmark testing of the improved function and will replace their current `gfprimck` by the optimized one in the next release of their Communications Toolbox.

5. CONCLUSION

Using course projects as a key component in a graduate course – in addition to traditional teaching methods – has shown to provide several advantages, but selecting suited project topics is crucial. The described course form can be more demanding for both students and instructor, requiring more interaction and participation on the students' part and teaching-on-demand for the instructor. The benefits observed are improved critical thinking and communication skills among students, as well as a boost in their selfconfidence in becoming good practical engineers. Because the topics are often new to the students, they are more likely to go new ways and come up with innovative solutions; the presented project work is in fact able to outperform a state-of-the-art software package.

6. ACKNOWLEDGMENT

The authors would like to thank The MathWorks for the interest shown in the improvements we made to their function, as well as the time and effort spent to answer emails and benchmark test the new function.

7. REFERENCES

[1] "The MathWorks." http://www.mathworks.com.

[2] L. Harte, *CDMA IS-95 for Cellular and PCS*. McGraw-Hill, 1999.

[3] D. E. Knuth, *The Art of Computer Programming, volume 2: Seminumerical Algorithms*. Reading, MA: Addison-Wesley, 3 ed., 1997.

[4] R. Lidl and H. Niederreiter, *Finite Fields*. Reading, MA: Addison-Wesley, 1983.

[5] P. Hellekalek, "Study of algorithms for primitive polynomials," report D5H-1, CEI-PACT Project, WP5.1.2.1.2, Research Institute for Software Technology, University of Salzburg, Austria, 1994.