

ECC 2nd homework

김동휘, 박태원, 송유석

CRC simulation (1 / 3)

- 목표 : CRC의 오류 검출 능력을 확인해보기

APPENDIX III
DATA FOR SOME REPRESENTATIVE CODES

Detection Capabilities	k_{\max}	$n-k$	$P(X)$	Reference
Any odd number of errors	any value	1	$1+X$	Theorem 2
Two errors, a burst of length 4 or less, 88 per cent of the bursts of length 5, 94 per cent of longer bursts*	11	4	$1+X+X^4$	Theorems 3, 5, 6
Two errors, a burst of 9 or less, 99.6 per cent of the bursts of length 10, 99.8 per cent of longer bursts	502	9	$1+X^4+X^9$	Theorems 3, 5, 6
Two bursts of length 2 or less, any odd number of errors, a burst of 5 or less, 93.8 per cent of the bursts of length 6, 96.9 per cent of longer bursts†	10	5	$(1+X+X^4)(1+X)=1+X^2+X^4+X^5$	Theorems 2, 5, 6, 7
Two bursts of combined length 12 or less, any odd number of errors, a burst of 22 or less, 99.99996 per cent of the bursts of length 23, 99.99998 per cent of longer bursts	22495	22	$(1+X^2+X^{11})(1+X^{11})=1+X^2+X^{13}+X^{22}$	Theorems 2, 5, 6, 8
Any combination of 6 or fewer errors, a burst of length 11 or less, 99.9 per cent of bursts of length 12, 99.95 per cent of longer bursts	12	11	$1+X^2+X^4+X^6+X^8+X^{10}+X^{11}$	Theorems 5, 6, and footnote 1
Any combination of 7 or fewer errors, any odd number of errors, a burst of length 31 or less, all but about 1 in 10^9 of longer bursts	992	31	$(1+X)(1+X^3+X^{10})$ $(1+X+X^2+X^3+X^{10})$ $(1+X^2+X^3+X^8+X^{10})$	Theorems 2, 5, 6, and footnotes 9, 12, 18

* Note: $1+X+X^4$ belongs to $e=15$ and $11+4=15$.

† Note: This is the code used in all examples.

CRC simulation (2 / 3)

1. CRC encode, decode 코드 작성하기

- 이번 과제에서 polynomial들은 오른쪽 예시와 같은 array로 표현함 e.g. $x^5 + x^4 + x^2 + 1 \rightarrow [1, 1, 0, 1, 0, 1]$
- encode 함수는 message를 받아서 codeword를 반환
- decode 함수는 receive를 받아서 오류 검출이 되면 1, 오류 검출이 안되면 0을 반환
- encode, decode 함수를 작성한 후에, test_functionality 함수를 통하여 encode, decode가 잘 되는지 확인

2. get_period 함수 구현하기

- $g(x) \mid x^e + 1$ 를 만족하는 e 의 최솟값을 구하는 코드 작성하기

3. Monte-Carlo simulation 코드 작성하기

- 주어진 iteration만큼 반복 (Iteration은 많이 할수록 더 정확한 결과를 얻을 수 있음).
- 발생하는 error type별로 발생한 횟수와 CRC가 해당 오류를 검출한 횟수를 측정하여 검출된 비율을 측정.
- test_functionality 코드를 참고해볼 것.
- 주의사항 : error를 랜덤으로 생성할 시, 에러 발생 확률은 0.5로 설정해야 함 (np.random.binomial 입력에서 p=0.5)
- 특정 한 error가 속하는 type이 여러 개인 경우에는 각각의 type에 대해 다 count 해야함.

CRC simulation (3 / 3)

- 실험할 generator polynomials : $x^5 + x^4 + x^2 + 1$, $x^8 + x^7 + x^6 + x^4 + x^2 + 1$
- 아래 그림과 같이 출력 (tqdm을 이용하여 실행 시간 측정하는 부분은 안 넣어도 됨)

```
Test the generator polynomial [1 1 0 1 0 1]
Period is 15

100%|██████████| 2000000/2000000 [01:22<00:00, 24121.11it/s]

Odd error(s)           : 100.00% [1000879 / 1000879]
Double errors          : 100.00% [6436 / 6436]
Burst error (length < n-k+1) : 100.00% [11752 / 11752]
Burst error (length = n-k+1) : 93.90% [8955 / 9537]
Burst error (length > n-k+1) : 96.87% [1916743 / 1978664]
```

예시 : $x^5 + x^4 + x^2 + 1$ 실험 결과

- 결과 캡처 사진과 결과에 대한 설명을 간단하게 1~2쪽으로 요약하여 정리하여 pdf 작성
- 제출 파일 : crc.py, report.pdf → [이름.tar](#) or [이름.tar.gz](#)
- 제출 방법
 1. `chmod o-rwx 이름.tar`
 2. `cp 이름.tar /home/ECC_ASSIGNMENTS/2nd/submit/`