Angriff auf Bivium mittels SAT Solver

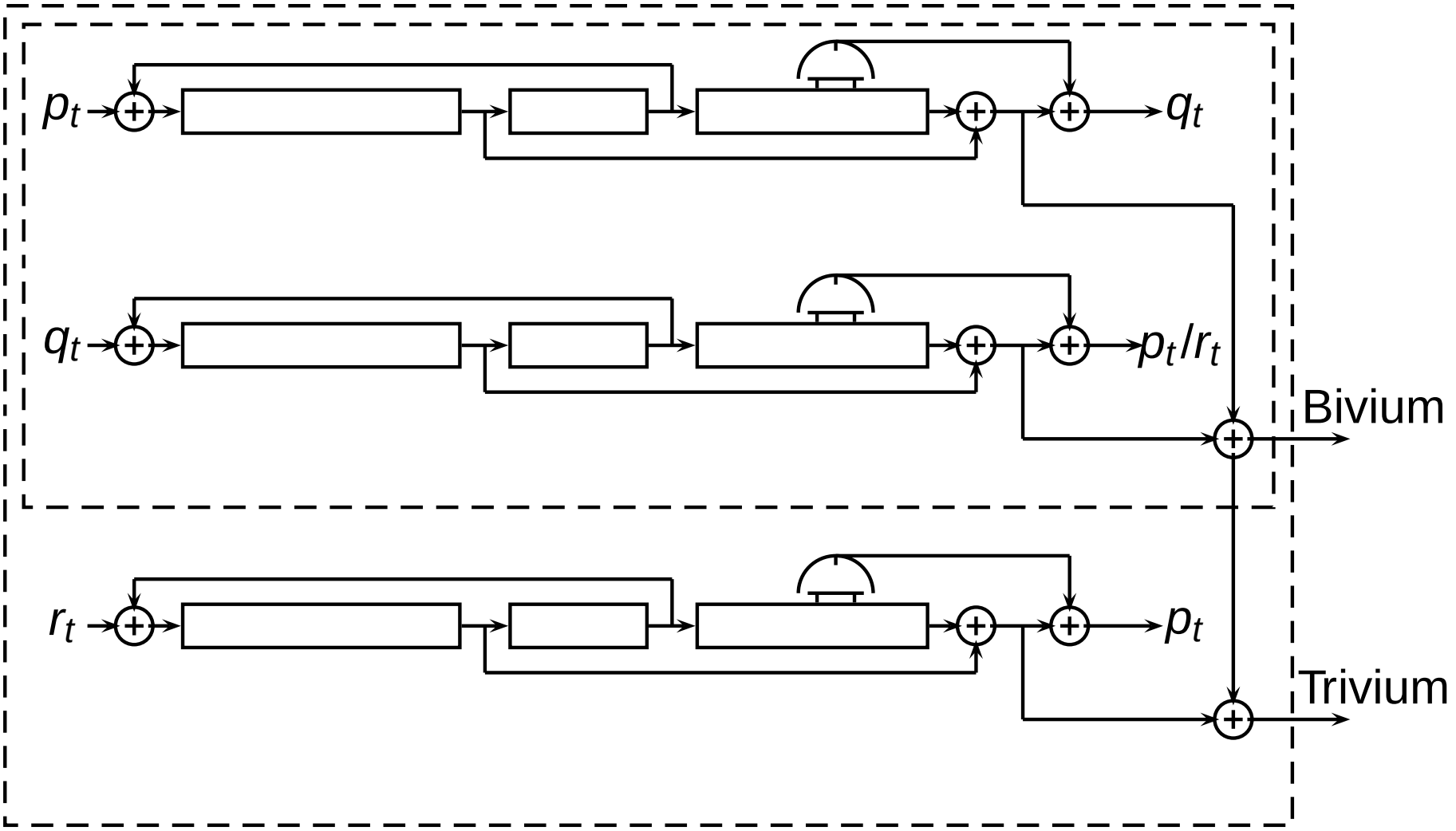
Tobias Eibach, Enrico Pilz

Fakultät für Informatik
Universität Ulm

9.11.2007

Outline

BIVIUM / TRIVIUM



GETTING EQUATIONS - 1

Algorithm 1 Bivium Pseudocode

FOR i from 1 to N do

$t_1 \leftarrow s_{66} + s_{93}$

$t_2 \leftarrow s_{162} + s_{177}$

$z_i \leftarrow t_1 + t_2$

$t_1 \leftarrow t_1 + s_{91} * s_{92} + s_{171}$

$t_2 \leftarrow t_2 + s_{175} * s_{176} + s_{69}$

$(s_1, s_2, \ldots, s_{93}) \leftarrow (t_2, s_1, \ldots, s_{92})$

$(s_{94}, s_{95}, \ldots, s_{177}) \leftarrow (t_1, s_{94}, \ldots, s_{176})$

Getting Equations - 2

$$s_{66} + s_{93} + s_{162} + s_{177} + z_1 = 0$$

$$s_{65} + s_{92} + s_{161} + s_{176} + z_2 = 0$$

$$\ldots$$

$$s_1 + s_{28} + s_{97} + s_{112} + z_{66} = 0$$

$$s_{27} + s_{69} + s_{96} + s_{111} + s_{162} + s_{175} * s_{176} + s_{177} + z_{67} = 0$$

$$s_{26} + s_{68} + s_{95} + s_{110} + s_{161} + s_{174} * s_{175} + s_{176} + z_{68} = 0$$

$$s_{25} + s_{67} + s_{94} + s_{109} + s_{160} + s_{173} * s_{174} + s_{175} + z_{69} = 0$$

$$\ldots$$

$$s_4 + s_{14} * s_{15} + s_{29} * s_{30} + s_{31} + s_{55} + s_{80} * s_{81} + s_{82} + s_{94} +$$

$$s_{95} * s_{96} + s_{97} + s_{122} * s_{123} + s_{124} + s_{160} + z_{147} = 0$$

$$\ldots$$

Moving to CNF

Some lines of a CNF file:

```
66 -93 -162 -177 0
-66 93 -162 -177 0
-66 -93 162 -177 0
-66 -93 -162 177 0
-178 66 93 171 91 92 0
-178 66 93 171 -91 92 0
-178 66 93 171 91 -92 0
178 -66 93 171 91 92 0
178 -66 93 171 -91 92 0
178 -66 93 171 91 -92 0
```

Bivium instances have about 445 variables and 9000 clauses.

Many Variations/Strategies

- How to split the 2 phases? (create CNF - solve CNF)
- When and how to split equations? More variables or higher degree?
- How many equations?
- Using Gaussian elimination?
- Also the following results imply certain strategies.

Current results

We studied several questions that come up when implementing the attack:

1. Which SAT solver to use?
2. Which variables to guess?
3. How many variables to guess?
4. What about the Hamming weight?
5. More ... but not in this talk.

Comparing SAT solvers

	guess 40	guess 45	guess 50
satelite	46.10	3.32	0.26
minisat	67.32	5.06	0.36
picosat	103.96	5.78	0.42
rsat	229.09	11.49	0.79
zchaff	735.08	17.36	0.78

Table: Comparing SAT solvers

(time for one instance, 100 random instances averaged, guess: Ending)

Where to guess

method	time
Beginning	204
Ending	9
Ending2	1070
DreiVier	60
Zufall1	791
Zufall2	263
Zufall3	2540

Table: Comparing different guessing strategies

(Time to solve 100 random instances, guessing 48 variables.)

TIME VS GUESS NUMBER

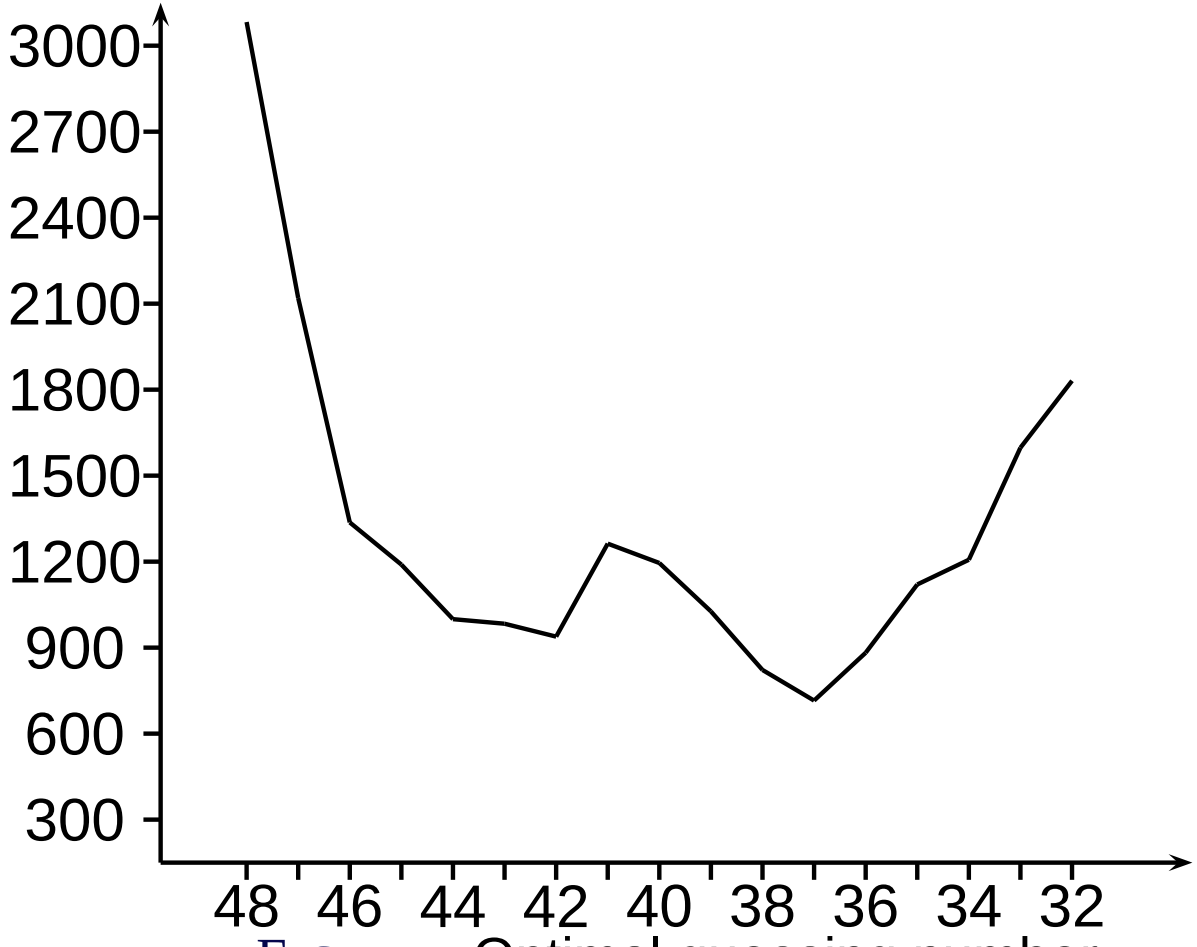


FIGURE: Optimal guessing number

(guess: Ending, 48 - 32 variables, time / 10^{10})

TIME VS HAMMING WEIGHT

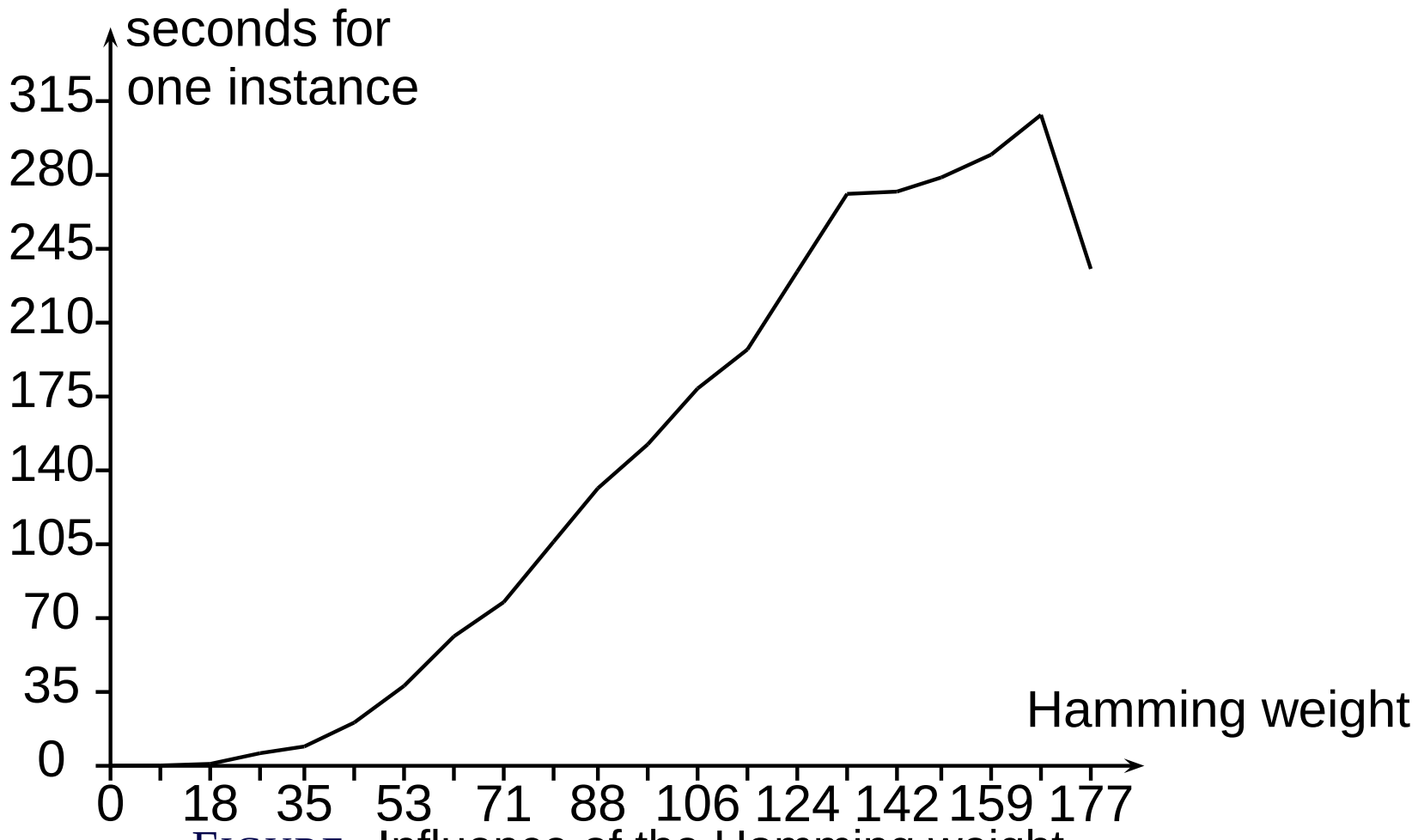


FIGURE: Influence of the Hamming weight

(guess: Ending - 36 variables, averaged over 100 instances)

Comparing to other attacks

Just to give a rough idea: (in seconds)

- Raddum: $\approx 2^{56}$ -> 7205759 E10
- Maximov: $\approx 2^{52.3}$ -> 554458 E10
- McDonald: guess 34 -> 440 -> total: 756 E10.
- Our current attack: guess 37 (Ending) -> 43.85 -> total: 603 E10.
- OBDDs ... ?
- Groebner basis / F4 / F5 ... ?

Outlook

Besides optimising this attack and producing more experimental results, the following should also be interesting:

- "Explaining" the experimental results
- Extending the results to Trivium
- Extending the approach to other stream ciphers
- Comparing the approach to other generic attacks

References

- Cannière and Preneel.
  TRIVIUM - a stream cipher construction inspired by block cipher design principles, 2005.
- Bard and Courtois and Jefferson.
  Efficient Methods for Conversion and Solution of Sparse Systems of Low-Degree Multivariat Polynomials over GF(2) via SAT-Solvers, 2007.
- Cameron McDonald and Chris Charnes and Josef Pieprzyk.
  Attacking Bivium with MiniSat, 2007.

The End

Thank you!

Questions?