ON THE DESIGN OF PERMUTATION P IN DES TYPE CRYPTOSYSTEMS

Lawrence Brown Jennifer Seberry

Department of Computer Science
University College, UNSW, Australian Defence Force Academy
Canberra ACT 2600. Australia

Abstract

This paper reviews some possible design criteria for the permutation P in a DES style cryptosystem. These permutations provide the diffusion component in a substitution-permutation network. Some empirical rules which seem to account for the derivation of the permutation used in the DES are first presented. Then it is noted that these permutations may be regarded as latin-squares which link the outputs of S-boxes to their inputs at the next stage. A subset of these with a regular structure, and which perform well in a dependency analysis are then presented. Some design rules are then derived, and it is suggested these be used to design permutations in future schemes for an extended version of the DES.

1. Introduction

The Data Encryption Standard (DES) [NBS77] is currently the only certified encryption standard. It has achieved wide utilization, particularly in the banking and electronic funds transfer areas, and is an Australian standard [ASA85] among others.

However at the time of its introduction, there was a considerable amount of controversy due both to the design criteria being classified, and to the choice of a 56-bit key that was considered too small [DiHe77], [Hell79]. However, on all other accounts DES appears to be an excellent cryptosystem. No short-cuts have been found to aid in the cryptanalysis of DES other than by exhaustive key-space search. With the current significant use of DES (especially in banking), there is interest in designing and building a DES-type cryptosystem with an extended key length.

In Brown [Brow88], the design criteria used in the DES, both those reported in the literature, and those noted by the author, were discussed. This paper further considers the design of the permutation box P , which provides the diffusion component of a substitution-permutation network in DES type cryptosystems. Such systems were originally devised by Shannon [Shan49], who termed this arrangement a **mixing transformation**. The S-boxes provided **confusion** of the input, and the P-boxes provided **diffusion** of the S-box outputs across the inputs to the next stage. Two key properties of such S-P networks are the **avalanche** property, identified by Feistel [Feis73]; and the **completeness** property, identified by Kam and Davida [KaDa79]. They ensure that every output bit becomes a function of each input bit in as few rounds as possible. Meyer [Meye78] (also in [MeMa82]) has quantified this property for the DES, by showing that after 5 rounds, every output bit is a function of all input bits. We have used this form of analysis as a measure of effectiveness in the design of permutation P . It is intended that as a result of this work, the design of the permutation P in an extended DES style scheme may be performed on a sounder theoretical basis.

2. Empirical P-box Design Criteria

The central component of the DES cryptosystem is the function g . This is implemented as a composition of an expansion function E which provides an autoclave function, eight substitution boxes (S-boxes) S which confuse the input bits, and then a permutation P which diffuses the outputs from each S-box to the inputs of a number of S-boxes in the next stage. A more detailed description of these functions may be found in [NBS77], [ASA85] or [SePi88]. For the purposes of this analysis, it is convenient to regard the DES as a mixing function, as detailed in Davio et al. [DDFG83], which emphasizes the analysis of the functional composition $P \cdot S \cdot E$ (see Fig 1.).

In Brown [Brow88], the following analysis of, and design rules for, permutation P were derived. This was done by analysing the $S \cdot P \cdot E$ functional composition, which forms one round of the S-P network in the DES:

$$R(i) = L(i-1) \oplus P(S(E(R(i-1)) \oplus K(i))).$$

The input of permutation P may be divided into 4-bit outputs from the eight S-boxes, and the output from E is divided into 6-bit inputs to the eight S-boxes at the next stage. Instead of expressing this permutation in terms of bit positions, it may be written in terms of which S-box output is connected by $P \cdot E$ to each input bit of the S-boxes at the next stage (see Table 1). Provided the S-boxes fulfil their design requirements, it may be assumed that all S-box outputs are equivalent, and that the inputs may be considered as three pairs ab , cd and ef . Note that due to the expansion function E , columns ab are identical to columns ef for the previous S-box.

Fig 1. DES as a Mixing Function

Table 1 - P.E Permutation							
S-box	Inputs from S-boxes						Excluded S-box
	a	b	c	d	e	f	
1	7	4	2	5	6	8	3
2	6	8	3	7	5	1	4
3	5	1	4	6	7	2	8
4	7	2	5	8	3	1	6
5	3	1	2	6	4	8	7
6	4	8	7	1	3	5	2
7	3	5	4	8	2	6	1
8	2	6	3	1	7	4	5

From this Brown [Brow88] derived the following set of empirical rules for designing permutation P :

- R1 Each of the S-box input bits $ab\ cd\ ef$ come from the outputs of different S-boxes.
- R2 None of the input bits $ab\ cd\ ef$ to a given S-box $S(i)$ comes from the output of that same S-box $S(i)$.
- R3 An output from $S(i-1)$ goes to one of the ef input bits of $S(i)$, and hence via E an output from $S(i-2)$ goes to one of the ab input bits.
- R4 An output from $S(i+1)$ goes to one of the cd input bits of $S(i)$.
- R5 For each S-box output, two bits go to ab or ef input bits, the other two go to cd input bits as noted in [Davi82].

These rules all appear consistent with the implementation of the avalanche and completeness effects by ensuring that every output bit becomes a function of each input bit in as few rounds as possible.

Permutations satisfying these rules have been generated. A total of 178 permutations were found, from 96 possible exclusion sets (the mandatory wirings required by rules 3 and 4 were arbitrarily assigned to bits c and e/a respectively). Each of these permutations could generate a number of possible permutations by swapping the order of bits in each of the pairs ab , cd , and ef ; or by rearranging the order of the four output bits from each S-box. However, Brown [Brow88] believes these variations are not significant, and notes that they do not change the result of the dependency analysis. Davies [Davi82] and Davio et al. [DDFG83] have also observed that a functionally equivalent SPE combination can be formed by rearranging the order of the S-box output bits (by rearranging columns), and by then altering permutation P to compensate.

To provide a measure of the effectiveness of the derived permutations, Meyer's analysis [Meye78], [MeMa82] of ciphertext dependence on plaintext bits was extended for the 178 permutations derived from these empirical rules. Briefly, following Meyer, this analysis may be described as follows. To provide a measure of this dependency, a 64×64 array $G_{a,b}$ is formed. Each element $G_{a,b}(i,j)$ specifies a dependency of output bit $X(j)$ on input bit $X(i)$, between rounds a and b . The number of marked elements in $G_{0,r}$ indicates the degree to which complete dependence was achieved by round r . Details of the derivation of this matrix, and the means by which entries are propagated, may be found in [MeMa82]. The extended analysis [Brow88] for the current DES, the 178 empirically generated permutations, and the worst case P (see Table 2), gave results as shown in columns 2, 3, and 7 of Table 3.

Table 2 - Worst Case DES P

1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5	6	6	6	6	7	7	7	7	8	8	8	8
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

3. Further Analysis of the Design of Permutation P

On closer examination, it was noted that these empirically derived rules for the design of permutation P , when written as in Table 1, actually result in a latin square. A **latin square** is a square of numbers in which each number occurs exactly once in each row and each column (see [DeKe74]). Note that the permutation P used in the current DES is not a latin square, however it may be transformed into one by selectively swapping some *ab/ef* (these being equivalent due to E) and *cd* pairs to align the highlighted values in Table 1 into the same columns. This transformation makes no difference to the dependency result obtained. Such a result cannot be accidental, and must be related to the desired properties of permutation P , namely to provide maximal diffusion among the S-boxes.

In order to explore this further, some possible permutations which could be used in a DES type system, and which form a latin-square when written as specified above, were generated. A sample space of 657096 such permutations were generated. These were analysed for effectiveness using Meyer's ciphertext dependence on plaintext, with results as summarised in column 4 of Table 3.

Table 3 - Dependency of Ciphertext on Plaintext Bits
for various DES permutations P by Round

Round	Current P	Empirical P	Sample Latin-Square P	Best Latin-Square P	Regular Form P	Worst P
1	6.25	6.25	6.25	6.25	6.25	6.25
2	32.06	32.03-32.10	30.49-32.20	32.02-32.23	30.47-32.23	21.09
3	73.49	73.44-73.58	70.36-73.78	73.44-73.83	70.31-73.83	43.75
4	96.90	96.87-96.95	95.34-97.05	96.88-97.07	95.31-97.07	68.75
5	100.00	100.00	100.00	100.00	100.00	89.06
6	100.00	100.00	100.00	100.00	100.00	98.44
7	100.00	100.00	100.00	100.00	100.00	100.00
8	100.00	100.00	100.00	100.00	100.00	100.00

The P-box used in the current DES has a propagation profile that falls fairly close to to the median of the 178 permutations generated by the empirical rules. It thus appears to be a fairly representative example of them. In turn, these also fall within the range found for the permutations derived from latin-squares, which obviously encompass the criteria needed to design them. In conjunction with the substantially inferior profile of the worst case P-box, this provides a strong indication that the design rules identified are comprehensive, at least as far as this aspect of the P-box design is concerned.

4. Analysis of the Best Latin-Square Permutations P

From the sample space of 657096 permutations, those permutations resulting in the highest percentage values in each round of the dependency analysis were extracted. A total of 20 such permutations were found. When examined, half of the columns (namely the *ab/ef* columns) of these permutations were found to be identical, with values as

given in Table 4. These columns provide inputs to two S-boxes due to expansion function E . Further, they were found to be nearly identical to the values generated by applying a difference function of

$$[-2 + 1 \ c \ d \ -1 + 2] \quad c, d \text{ arbitrary} \quad (1)$$

to the input S-box number, as shown in full in Table 4.

Table 4 - Fixed Columns in Sample Permutations P
Replicated pattern in the best sample permutations
3 . . 7 8 . . 1 4 . . 5 2 . . 3 6 . . 4 7 . . 8 5 . . 6 1 . . 2
Pattern generated by a $[-2 + 1 \ c \ d \ -1 + 2]$ difference function
2 . . 8 3 . . 1 4 . . 2 5 . . 3 6 . . 4 7 . . 5 8 . . 6 1 . . 7

From this, we suggested that permutations with a fixed *ablef* column structure whose values are those generated by difference function (1) may perform well. Permutations of this form were generated, a total of 264 being found. The results of the dependency analysis on these permutations is given in column 5 of Table 3. Those providing the best results in this analysis were extracted, 100 such permutations being found. These were found to be grouped into pairs, with only columns c and d interchanged. These pairs in turn were grouped by exclusion set (the set of values not used as inputs to each S-box). The only difference between them is the swapping of some of the cd bits, a transformation which makes no difference to the dependency results. A total of 18 exclusion sets were found among the best permutations, and a sample from each is given in Table 5.

Table 5 - Best Latin-Square Permutations P		
Sample Permutation	Exclusion Set	Number
25683671478258136124723583461457	45678123	2
25683751468251736824721583461437	46781325	4
25683751467251836824721583461437	46871325	8
25683751467251836814723584261347	46872135	8
25683651478251736824731582461437	47681235	4
25683651478251736814723584261347	47682135	8
24683751468258736124721583461537	56718324	4
24683751468251736814723583261547	56782143	4
24683751467251836814732582461537	56872134	4
24683651478258736124731582461537	57618234	4
24683651478258736124731582361547	57618243	8
24683651478251736814723583261547	57682143	8
24583761468258736124731582361547	65718243	8
24583761468258736124721583461537	65718324	8
24583761468251736824731582361547	65781243	4
24583761467251836824731582461537	65871234	4
24583761467251836824721583461537	65871324	8
24583561467257836814712582361347	67812345	2

We noted that among these permutations are two which may be generated with difference functions

$$[-2 +1 +4 -3 -1 +2], \quad \text{and} \quad (2)$$

$$[-2 +1 +3 +4 -1 +2] \quad (3)$$

being the first and last entries in Table 5 respectively.

5. A Regular Form for Permutation P

As further confirmation of this result, all permutations P which may be generated by a regular difference function on the target S-box were constructed, 120 being found. Their cipher-plaintext bit dependency propagation is given in Table 3. The 8 best of these are indeed the regular permutations formed using the difference functions (2) and (3), and their equivalents formed by swapping *ab/ef* or *cd* columns. They are listed in Table 6. These permutations would thus seem to be excellent candidates for any reimplementation of the DES using 64-bit blocks. However, some problems are noted when the dependency results are examined further, as indicated below.

Table 6 - Best Regular Form Permutations P	
Permutation	Difference Fn
74538564167527863817412852316342	-2 +3 +4 +2
75438654176528763187421853216432	-2 +4 +3 +2
75638674178528163127423853416452	-2 +4 -3 +2
76538764187521863217432854316542	-2 -3 +4 +2
24583561467257836814712582361347	+1 +3 +4 -1
25483651476258736184721583261437	+1 +4 +3 -1
25683671478258136124723583461457	+1 +4 -3 -1
26583761487251836214732584361547	+1 -3 +4 -1

6. A Further Criterion for Best Permutations

In the analysis so far, the measure of effectiveness being used is the rate of growth of dependence of output bits on input bits, using Meyer's analysis. However this dependence may be further characterized, by a dependence through message inputs (bits *bcd*e) only, autoclave inputs (bits *a*f) only, or through both. If, as a measure of effectiveness, we examine the rate of growth of dependence on input bits through both autoclave and message bits, this demonstrates that a strictly regular permutation is not necessarily the most effective (see Table 7). Rather, the best latin-square permutations, that is with only two of four columns fixed, performs better by this criterion. Also, the original DES performs better still, but at a price of lower growth in overall dependence. There thus, seems to be some tradeoff between regularity, and best performance.

In the extended scheme being developed, the choice of some of the best latin square permutations, would thus seem to be indicated. In subsequent work on permutation PC2 and the key schedule, a subset of these, which also perform best using a measure of dependence of output bits on key bits have been found. These are listed in Table 8, and were used to produce the results in Table 7.

**Table 7 - Dependency of Ciphertext bits on
Plaintext bits via Message, Autoclave and Both Inputs**

Round	Current P				Best Latin Square P				Best Regular P			
	Msg	Auto	Both	Total	Msg	Auto	Both	Total	Msg	Auto	Both	Total
1	192	64	0	6.25	192	64	0	6.25	192	64	0	6.25
2	641	499	173	32.06	656	512	152	32.23	616	512	192	32.23
3	834	806	1370	73.49	864	864	1296	73.83	784	960	1280	73.83
4	353	371	3245	96.90	368	448	3160	97.07	328	640	3008	97.07
5	0	0	4096	100.00	0	32	4064	100.00	0	128	3968	100.00
6	0	0	4096	100.00	0	0	4096	100.00	0	0	4096	100.00

**Table 8 - Best Latin Square Permutations P
by both Ciphertext-Plaintext and Ciphertext-Key Dependence**

2 5 6 8 3 6 5 1 4 7 8 2 5 1 7 3 6 8 1 4 7 2 3 5 8 4 2 6 1 3 4 7
2 5 6 8 3 6 5 1 4 7 8 2 5 1 7 3 6 8 1 4 7 3 2 5 8 2 4 6 1 4 3 7
2 5 6 8 3 6 5 1 4 8 7 2 5 7 1 3 6 1 8 4 7 2 3 5 8 4 2 6 1 3 4 7
2 5 6 8 3 6 5 1 4 8 7 2 5 7 1 3 6 1 8 4 7 3 2 5 8 2 4 6 1 4 3 7
2 6 5 8 3 5 6 1 4 7 8 2 5 1 7 3 6 8 1 4 7 2 3 5 8 4 2 6 1 3 4 7
2 6 5 8 3 5 6 1 4 7 8 2 5 1 7 3 6 8 1 4 7 3 2 5 8 2 4 6 1 4 3 7
2 6 5 8 3 5 6 1 4 8 7 2 5 7 1 3 6 1 8 4 7 2 3 5 8 4 2 6 1 3 4 7
2 6 5 8 3 5 6 1 4 8 7 2 5 7 1 3 6 1 8 4 7 3 2 5 8 2 4 6 1 4 3 7

7. Design Rules for Permutation P

In the light of these experiments, the current set of rules for designing permutation P in a DES like scheme are:

- R1 Each of the input bits $a b c d e f$ for S-box $S(i)$ must come from the outputs of different S-boxes in the previous round.
- R2 None of the input bits $a b c d e f$ for S-box $S(i)$ may come from the output of that same S-box $S(i)$ in the previous round.
- R3 For each input bit $a, b, c, d, e,$ or f over all of the S-boxes, the S-boxes permuted to that bit must differ (ie each column forms a permutation of integers 1 to 8).
- R4 An output from $S(i-2)$ goes to one of the ab input bits of $S(i)$, and from $S(i+1)$ to the other. Hence via E an output from $S(i-1)$ goes to one of the ef input bits, and from $S(i+2)$ to other.
- R5 Outputs from two of $S(i-3), S(i+3),$ and $S(i+4)$ go to cd input bits of $S(i)$.
- R6 For each S-box output, two bits go to ab or ef input bits, the other two go to cd input bits as noted in [Davi82].

Rules R1, R2 and R3 constrain the permutation to be a latin-rectangle, whilst the remaining rules further constrain the permutation to select outputs which maximize the growth of bit dependence. Having used these rules to construct a permutation, the dependency analysis may then be used to verify its effectiveness.

8. Regular Permutations P in An Extended DES

Permutations with strictly regular form were generated for an extended DES using 128-bit blocks. The 4 best of these were produced using a difference function

$$[-2 +1 -7 +7 -1 +2] \quad (4)$$

and its equivalents by column swapping, as shown in Table 9. The cipher-plaintext dependency results are given in Table 10 (along with those for a worst case, and sample permutations given in Brown [Brow88]). These best permutations alone of all those generated, resulted in complete dependency of ciphertext bits on all plaintext bits in 5 rounds, the same as for the current size scheme. Work is continuing to derive the best permutations for use in the extended scheme when dependence on both message and autoclave inputs is considered.

Table 9 - Best Regular Form Extended Permutations P

2 10 8 16 3 11 9 1 4 12 10 2 5 13 11 3 6 14 12 4 7 15 13 5 8 16 14 6 9 1 15 7 10 2 16 8 11 3 19 12 4 2 10 13 5 3 11 14 6 4 12 15 7 5 13 16 8 6 14 1 9 7 15
2 8 10 16 3 9 11 1 4 10 12 2 5 11 13 3 6 12 14 4 7 13 15 5 8 14 16 6 9 15 1 7 10 16 2 8 11 1 3 9 12 2 4 10 13 3 5 11 14 4 6 12 15 5 7 13 16 6 8 14 1 7 9 15
15 10 8 3 16 11 9 4 1 12 10 5 2 13 11 6 3 14 12 7 4 15 13 8 5 16 14 9 6 1 15 10 7 2 16 11 8 3 1 12 9 4 2 13 10 5 3 14 11 6 4 15 12 7 5 16 13 8 6 1 14 9 7 2
15 8 10 3 16 9 11 4 1 10 12 5 2 11 13 6 3 12 14 7 4 13 15 8 5 14 16 9 6 15 1 10 7 16 2 11 8 1 3 12 9 2 4 13 10 3 5 14 11 4 6 15 12 5 7 16 13 6 8 1 14 7 9 2

Table 10 - Dependency of Ciphertext bits on
Plaintext bits in Extended (128-bit) DES

Round	Best Regular Form P	Sample XDES P	Worst XDES P
1	3.13	3.125	3.125
2	17.58	17.68	10.55
3	52.34	50.98	21.88
4	87.50	84.47	34.38
5	100.00	98.44	46.88
6	100.00	100.00	59.38
7	100.00	100.00	71.88
8	100.00	100.00	84.38
9	100.00	100.00	94.53
10	100.00	100.00	99.21
11	100.00	100.00	100.00

9. Conclusion

This paper reviews some possible design criteria for the permutation P in a DES style cryptosystem. These permutations provide the diffusion component in a substitution-permutation network. Some empirical rules which seem to account for the derivation of the permutation used in the DES are first presented. Then it is noted that these permutations may be regarded as latin-squares which link the outputs of S-boxes to their inputs at the next stage. When a sample of these latin-square permutations were generated, they were found to span those generated using the empirical rules. The best of these permutations suggested that permutations having a fixed column structure as generated by difference function (1) would perform well. Permutations with this structure were generated, and the best of these analysed. They were found to include the

permutations generated using difference functions (2) and (3), which were strictly regular. Whilst these performed well in growth of overall bit dependence, when dependence on both message and autoclave bits were examined, these were found to be slightly less than optimal. From these a set of design rules have been developed for permutation P. An extension of these results to an extended DES was then analysed, and 4 permutations formed by difference function (4) were found to result in complete ciphertext dependency on plaintext bits after 5 rounds, a result that is the same as with the current size scheme.

Acknowledgements

To the following members of the Centre for Computer Security Research: Leisa Condie, Thomas Hardjono, Arthur Lagos, Mike Newberry, Cathy Newberry, Josef Pieprzyk, and Jennifer Seberry; and to: Dr. George Gerrity, Dr. Andrez Goscinski, and Dr. Charles Newton; for their comments on, suggestions about, and critiques of this paper. Thankyou.

References

- [ASA85] ASA, "*Electronics Funds Transfer - Requirements for Interfaces, Part 5, Data Encryption Algorithm*," AS2805.5-1985, Standards Association of Australia, Sydney, Australia, 1985.
- [Brow88] L. Brown, "A Proposed Design for an Extended DES," in *Proc. Fifth International Conference and Exhibition on Computer Security*, IFIP, Gold Coast, Queensland, Australia, 19-21 May, 1988.
- [Davi82] D. W. Davies, "Some Regular Properties of the Data Encryption Standard," in *Advances in Cryptology - Proc. of Crypto 82*, D. Chaum, R. L. Rivest and A. T. Sherman (editors), pp. 89-96, Plenum Press, New York, Aug. 23-25, 1982.
- [DDFG83] M. Davio, Y. Desmedt, M. Fosseprez, R. Govaerts, J. Hulsbosch, P. Neutjens, P. Piret, J. Quisquater, J. Vanderwalle and P. Wouters, "Analytical Characteristics of the DES," in *Advances in Cryptology - Proc. of Crypto 83*, D. Chaum, R. L. Rivest and A. T. Sherman (editors), pp. 171-202, Plenum Press, New York, Aug. 22-24, 1983.
- [DeKe74] J. Denes and A. D. Keedwell, *Latin Squares and their Applications*, English Universities Press Limited, London UK, 1974.
- [DiHe77] W. Diffie and M. E. Hellman, "Exhaustive Cryptanalysis of the NBS Data Encryption Standard," *Computer*, vol. 10, no. 6, pp. 74-84, June 1977.
- [Feis73] H. Feistel, "Cryptography and Computer Privacy," *Scientific American*, vol. 228, no. 5, pp. 15-23, May 1973.
- [Hell79] M. E. Hellman, "DES will be totally insecure within ten years," *SPECTRUM*, vol. 16, no. 7, pp. 31-41, July 1979. With rebuttals from George I. Davida, Walter Tuchman, & Dennis Branstad.
- [KaDa79] J. B. Kam and G. I. Davida, "Structured Design of Substitution-Permutation Encryption Networks," *IEEE Trans. on Computers*, vol. C-28, no. 10, pp. 747-753, Oct. 1979.

- [Meye78] C. H. Meyer, "Ciphertext/plaintext and ciphertext/key dependence vs number of rounds for the data encryption standard," in *AFIPS Conf. Proc.* 47, pp. 1119-1126, AFIPS Press, Montvale NJ, USA, June 1978.
- [MeMa82] C. H. Meyer and S. M. Matyas, *Cryptography: A New Dimension in Data Security*, John Wiley & Sons, New York, 1982.
- [NBS77] NBS, "*Data Encryption Standard (DES)*," FIPS PUB 46, US National Bureau of Standards, Washington, DC, Jan. 1977.
- [SePi88] J. Seberry and J. Pieprzyk, *Cryptography: An Introduction to Computer Security*, Prentice Hall, Englewood Cliffs, NJ, 1988.
- [Shan49] C. E. Shannon, "Communication Theory of Secrecy Systems," *Bell System Technical Journal*, vol. 28, no. 10, pp. 656-715, Oct. 1949.