On the Design of Permutation P in DES Type Cryptosystems

Lawrence Brown and 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 an extremely regular structure, and which perform well in a dependency analysis are then presented and suggested for use in future schemes of both current and extended versions 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 both due to the classification of its design criteria, and with the choice of a 56-bit key as being too small [DiH77], [Hel79]. 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 [Bro89], 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 original devised by Shannon [Sha49], 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 [Fei73]; and the **completeness** property, identified by Kam and Davida [KaD79]. They ensure that every output bit becomes a function of each input bit in as few rounds as possible. Meyer [Mey78] (also in [MeM82]) has quantified this property for the DES, by showing that after 5 rounds, every output bit is a function of all input bits. The authors 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. Current P-box Design Criteria

The central component of the DES cryptosystem is the function g. It is 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 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 [SeP89]. For the purposes of this analysis, it is convenient to regard the DES as a 48-bit mixing function, as detailed in Davio et al [DDF83], which emphasizes the analysis of the functional composition $P.S.E$. In Brown [Bro89], the following analysis of, and design rules for, permutation P were derived by analysing this $S.P.E$ functional composition, which forms one round of the S-P network:

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

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 $P.E$ 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*.

Table 1 - P.E Permutation

S-box	Inputs from S-boxes a	b	c	d	e	f	Excluded S-box
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 the authors noted that permutation P ensures that:

• 1 each of the S-box input bits *ab cd ef* come from the outputs of different S-boxes.

• 2 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).

• 3 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.

• 4 an output from S(i+1) goes to one of the *cd* input bits of S(i).

• 5 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 [Dav82].

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 criteria 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, the authors believe these variations are not significant, and note that they do not change the result of the dependency analysis. Davies [Dav82] and Davio et al [DDF83] have also observed that a functionly equivalent $S.P.E$ 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 [Mey78], [MeM82] of ciphertext dependence on plaintext bits was extended for these newly derived permutations. 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 $G_{a,b}(i, j)$ of which 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 [MeM82]. The authors repeated this analysis for the current DES, and extended it by analysing each of the 178 empirically generated permutations, as well as the worst case P (see Table 2), with results as shown in columns 2, 3, and 6 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. 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* 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. A **latin square** is a square of numbers in which each number occurs exactly once in each row and each column (see [DeK74]). 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 summarized in column 4 of Table 3.

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 my 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 resulting in the highest percentage values in the dependency analysis were extracted, a total of 20. 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

$$[-2 + 1\ c\ d - 1 + 2] \qquad c,\ d\ arbitary \tag{4.1}$$

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, the authors suggested that permutations with a fixed *ab*/*ef* column structure whose values are those generated by difference function (4.1) 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 were extracted, 100 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.

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

$$[-2 + 1 + 4 - 3 - 1 + 2], \quad \text{and} \tag{4.2}$$

$$[-2 + 1 + 3 + 4 - 1 + 2] \tag{4.3}$$

being the first and last entries in Table 5 respectively. These would, the authors believe, be a good choice for implementation in a practical scheme because of their regular structure.

5. A Regular Form for Permutation P

As further confirmation of this result, all permutations P which have the form of a difference function on the target S-box were constructed, 120 being found. There cipher-plaintext bit dependency propogation is given in Table 3. The 8 best of these are indeed the regular permutations formed using the difference functions (4.2) and (4.3), 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.

Table 6 - Best Regular Form Permutations P
7 4 5 3 8 5 6 4 1 6 7 5 2 7 8 6 3 8 1 7 4 1 2 8 5 2 3 1 6 3 4 2
7 5 4 3 8 6 5 4 1 7 6 5 2 8 7 6 3 1 8 7 4 2 1 8 5 3 2 1 6 4 3 2
7 5 6 3 8 6 7 4 1 7 8 5 2 8 1 6 3 1 2 7 4 2 3 8 5 3 4 1 6 4 5 2
7 6 5 3 8 7 6 4 1 8 7 5 2 1 8 6 3 2 1 7 4 3 2 8 5 4 3 1 6 5 4 2
2 4 5 8 3 5 6 1 4 6 7 2 5 7 8 3 6 8 1 4 7 1 2 5 8 2 3 6 1 3 4 7
2 5 4 8 3 6 5 1 4 7 6 2 5 8 7 3 6 1 8 4 7 2 1 5 8 3 2 6 1 4 3 7
2 5 6 8 3 6 7 1 4 7 8 2 5 8 1 3 6 1 2 4 7 2 3 5 8 3 4 6 1 4 5 7
2 6 5 8 3 7 6 1 4 8 7 2 5 1 8 3 6 2 1 4 7 3 2 5 8 4 3 6 1 5 4 7

6. Regular Permutations P in An Extended DES

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

$$[-2 + 1 - 7 + 7 - 1 + 2] \tag{6.1}$$

and its equivalents by column swapping, as shown in Table 7. The cipher-plaintext dependency results are given in Table 8 (along with those for a worst case, and sample permutations given in a previous paper [Bro89]). These best permutations, and they alone, resulted in complete dependency of ciphertext bits on all plaintext bits in 5 rounds, the same as for the current size scheme. Thus these permutations would seem to be excellent candidates for use in our proposed extended scheme.

7. Conclusion

A set of empirical design rules for the permutaion P in a DES style cryptosystem is descibed. It is then noted that permutations generated according to these rules are all latin-squares. 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 (4.1) would perform optimally. Permutations with this structure were generated, and the best of these analysed. They were found to include the permutations generated using difference functions (4.2) and (4.3), which are suggested for use in practical DES type schemes because of their regularity. An extension of these results to an extended DES was then analysed, and 4 permutations formed by difference function (6.1) 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. Such permutations are thus suggested for use in any extended schemes.

Table 7 - 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 1 9 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

Acknowledgements

To the members of the Computer Security Research (Crypto) Group, and the other staff in the Department, for their comments on this paper.

References

[ASA85] ASA, "*Electronics Funds Transfer - Requirements for Interfaces, Part 5, Data Encryption Algorithm*," AS2805.5-1985, Standards Association of Australia, Sydney, Australia, 1985.

[Bro89] L. Brown, "A Proposed Design for an Extended DES," in *Computer Security in the Age of Information*, W. J. Caelli (editor), North-Holland, Amsterdam, 1989.

[Dav82] 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, |August.| 23-25, 1982.

[DDF83] 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, |August.| 22-24, 1983.

[DeK74] J. Denes and A. D. Keedwell, *Latin Squares and their Applications*, English Universities Press Limited, London UK, 1974.

[DiH77] W. Diffie and M. E. Hellman, "Exhaustive Cryptanalysis of the NBS Data Encryption Standard," *Computer*, vol. 10, no. 6, pp. 74-84, |June| 1977.

[Fei73] H. Feistel, "Cryptography and Computer Privacy," *Scientific American*, vol. 228, no. 5, pp. 15-23, |May| 1973.

[Hel79] 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, and Dennis Branstad.

[KaD79] 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, |October.| 1979.

[Mey78] 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.

[MeM82] 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, |January.| 1977.

[SeP89] J. Seberry and J. Pieprzyk, *Cryptography: An Introduction to Computer Security*, |Prentice Hall, Englewood Cliffs, NJ|, 1989.

[Sha49] C. E. Shannon, "Communication Theory of Secrecy Systems," *Bell System Technical Journal*, vol. 28, no. 10, pp. 656-715, |October.| 1949.