**Haute école Spécialisée Bernoise**
**Haute école Technique et Informatique**

# Action ciphers – A new core component for E/D similar block ciphers

Dr. David-Olivier Jaquet-Chiffelle

Rapport de recherche No. 1 du 13 mars 2006

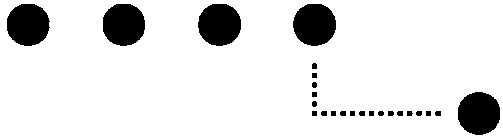
La série « Rapports de recherche de la Haute école Technique et Informatique » publie des connaissances et des résultats issus de la recherche et du développement de la Haute école Technique et Informatique HTI.

**Éditeur**
Haute école Technique et Informatique HTI, Case Postale, CH-2501 Bienne / Biel
Recherche et Développement, Faubourg du Lac 103b
T +41 (0)32 321 64 64,  F +41 (0)32 321 65 65
technologietransfer@hti.bfh.ch
www.hti.bfh.ch/recherche

**Concept et réalisation**
Jens Geisel

**Comité de lecture**
Dr. David-Olivier Jaquet-Chiffelle (rapport) et Jens Geisel (pages de titre)

**Rédaction**
Dr. David-Olivier Jaquet-Chiffelle (rapport) et Jens Geisel (pages de titre)



**Haute école Spécialisée Bernoise**
**Haute école Technique et Informatique**

# Action ciphers – A new core component for E/D similar block ciphers

Dr. David-Olivier Jaquet-Chiffelle

Rapport de recherche No. 1 du 13 mars 2006

**Abstract**

In cryptography, most practical block ciphers are E/D similar. Three core components are usually considered and combined to build E/D similar iterated block ciphers. In this report, we define a new one —the action ciphers— that extends the classical concept of group ciphers.Then we investigate a subset of action ciphers, the so-called σ-action ciphers, and give a cryptological characterization for this subset. All group ciphers are σ-action ciphers as well, but choosing a σ-action cipher which is not a group cipher eliminates “bad'” properties of group ciphers while keeping the “good” ones. Any group cipher component in any block cipher can be replaced by a σ-action cipher; such a replacement has no impact on the key-schedule algorithm.Finally, in the appendix, we show an application of these concepts that goes beyond the original scope of core components in E/D similar iterated block ciphers. Indeed we replace some group operations in IDEA by σ-action operations and get a large family of IDEA-like algorithms which can be customized.

**Key words**

cryptography, E/D similar block ciphers, group ciphers, action ciphers, IDEA

Bienne / Biel, le 13 mars 2006

# Author

**Dr. David-Olivier Jaquet-Chiffelle**, Professor of Mathematics and Cryptology at the University of applied Sciences of Berne, Founder and Head of V.I.P – Virtual Identity and Privacy research center. Lecturer at ESC (School of Criminal Sciences) at the University of Lausanne, Switzerland.

University of applied Sciences of Berne, School of Engineering and Information Technology, Rue de la Source 21, CP 1180, CH-2501 Bienne

+41 (0)32 321 62 66, david-olivier.jaquet-chiffelle@bfh.ch, www.vip.ch

**Description of V.I.P – Virtual Identity and Privacy research center**

Our group integrates a wide range of skills in computer science and applied mathematics. We cover subjects that deal with security and privacy, cryptology, identities and virtual identities, PETs (privacy enhancing technologies), pseudonyms, anonymization and data mining techniques as well as applied statistics in sensitive environments (for example in the medical sector). Our work is very often interdisciplinary.

VIP is part of “Technology for human beings”, a HTI pole of research; it is also member of EEMA, as well as of the “Security and Privacy” pole of ICTnet in Switzerland.

VIP belongs to the FIDIS consortium. FIDIS (Future of Identity in the Information Society) is a NoE (Network of Excellence) supported by the European Union under the 6th Framework Program for Research and Technological Development within the Information Society Technologies (IST) priority in the Action Line: “Towards a global dependability and security framework”.

# Contents

# Action ciphers — A new core component for E/D similar block ciphers

Dr. David-Olivier Jaquet-Chiffelle
*University of applied Sciences of Berne*

**Abstract** - *In cryptography, most practical block ciphers are E/D similar. Three core components are usually considered and combined to build E/D similar iterated block ciphers. In this report, we define a new one —the action ciphers— that extends the classical concept of group ciphers.*
*Then we investigate a subset of action ciphers, the so-called $\sigma$-action ciphers, and give a cryptological characterization for this subset. All group ciphers are $\sigma$-action ciphers as well, but choosing a $\sigma$-action cipher which is not a group cipher eliminates "bad" properties of group ciphers while keeping the "good" ones...*
*Any group cipher component in any block cipher can be replaced by a $\sigma$-action cipher ; such a replacement has no impact on the key-schedule algorithm.*
*Finally, in the appendix, we show an application of these concepts that goes beyond the original scope of core components in E/D similar iterated block ciphers. Indeed we replace some group operations in IDEA by $\sigma$-action operations and get a large family of IDEA-like algorithms which can be customized.*

Keywords: cryptography, E/D similar block ciphers, group ciphers, action ciphers, IDEA.

## 1 Introduction

An Encryption/Decryption similar (E/D similar) block cipher is a block cipher for which the encryption and the decryption process are similar : decryption can be done by modifying only the key-schedule algorithm. The same hardware realization can be used for both processes. The advantages in term of cost and convenience of using the same hardware for both the encryption and the decryption certainly explains why E/D similar algorithms are so popular. Most practical block ciphers are E/D similar.

In his Ph.D. thesis, X. Lai describes four constructions of E/D similar iterated block ciphers. They are based on three core components which he then combines. The three core components are

- involutary permutations,
- involution ciphers and
- group ciphers.

Involutary permutations are permutations of order 2 ; they are key-independant.

An involution cipher is any cipher for which the encryption process is exactly the same as the decryption process ; in other words, encrypting twice (with the same key) leads to the identity transformation : any plaintext remains unchanged.

Group ciphers will be discussed in section 2. These ciphers provide perfect secrecy for uniformly random one-time keys. However, a group is a rich mathematical structure with (too) many algebraic properties. To prove that a group cipher is E/D similar, we need only a few of those properties, the "good" ones. The unnecessary properties — the "bad" ones — give extra tools to the enemy cryptanalyst. For example, in a group cipher, the ciphering operation is associative ; it is even commutative when the group is Abelian. Those unneeded mathematical properties help the enemy cryptanalyst and, therefore, should be avoided from a cryptological point of view.

In this report, we define a new core component for E/D similar block ciphers — action ciphers — which extends the concept of group ciphers. As we will see, not all action ciphers are good cryptographically speaking. Only faithful actions can lead to action ciphers that provide perfect secrecy for uniformly random one-time keys.

The so-called $\sigma$-action ciphers (a special type of action ciphers) are excellent cryptographically speaking. They keep all the good properties of a group cipher ; they provide in particular perfect secrecy for uniformly random one-time keys. Actually, all group ciphers are also $\sigma$-action ciphers but group ciphers are the only $\sigma$-action ciphers for which the ciphering operation is associative. Indeed, in a $\sigma$-action cipher, the ciphering operation is neither associative nor commutative except if it is a group cipher, even if the action comes from an Abelian group. From a cryptological point of view, this is a significant improvement in comparison to group ciphers.

Any group cipher component of an E/D similar iterated block cipher can be replaced by a $\sigma$-action cipher. Moreover this substitution keeps the original key-schedule algorithm.

Last but not least, the group operations used in IDEA satisfy some partial distributive laws ([4]). These partial distributive laws have been used to attack IDEA with three rounds. Even though there has been no evidence that this poses a practical threat for IDEA with eight rounds, the destruction of these partial ditributive laws would increase the internal security. Replacing some of the group ciphers in IDEA by $\sigma$-action ciphers can destroy these partial distributive laws and consequently make the core operations in IDEA even more incompatible with each other.

# 2 Group ciphers

As we consider E/D similar block ciphers, the same set $X$ describes the set of all possible plaintexts and the set of all possible ciphertexts. Note that there is no need to restrict $X$ to ${\mathbb{F}_2}^m$.

Let's consider $G$ the set of all possible keys. When $G = X$ is a finite group for the operation $*$, we define the corresponding group cipher as the translation of any plaintext $x$ by the key $g \in G$. There is no need to restrict $G$ to 2-Sylow groups or to Abelian groups.

The translation can be either to the left (left group cipher), i.e. the ciphertext $y$ is equal to $g * x$, or to the right (right group cipher) $y = x * g$. When the group is Abelian, left and right group ciphers are the same. In the following, we will only consider left group ciphers. The theory and the properties for right group ciphers are similar.

In order to prove that encryption and decryption are similar in a group cipher, we only use the following fundamental property :

**Fundamental property** of a group cipher :

If $y = g * x$ then
$x = g^{-1} * y$ where $g^{-1}$ is the inverse of $g$ in $G$.

In other words, if we use the key $g$ to encrypt a plaintext $x$, we will use the same transformation, but with the key $g^{-1}$, to recover $x$ from the ciphertext.

This fundamental property leads to the key-schedule algorithm for decryption given the key-schedule algorithm for encryption in E/D similar iterated block ciphers using group cipher components. If the subkeys used in the group cipher components during encryption are $k_1, k_2, \ldots, k_r$ then the subkeys used during decryption are $k_r^{-1}, k_{r-1}^{-1}, \ldots, k_1^{-1}$.

# 3 Action ciphers

In this section we introduce a generalization of the concept of group ciphers : the action ciphers.

We keep the same notation as above ; now the set $X$ can be different from $G$ and does not need to have a group structure. Suppose that $G, *$ is a finite group and that $\alpha$ is a left action of $G$ on $X$. We will use the symbol $\cdot$ to represent the action operation :

$$\alpha : G \times X \longrightarrow X, \quad \alpha(g\,; x) = g \cdot x$$

We define the corresponding action cipher in the following way :

**Definition 3.1** *With the above notation, the (left) action cipher corresponding to $\alpha$ is*

*the cipher which transforms any plaintext $x$ into the ciphertext $y = g \cdot x$, where $g$ is the key.*

A left (resp. right) action cipher comes from a left (resp. right) action. In this paper, we consider only left action ciphers. The theory and the properties for right action ciphers are similar.

By definition, a (left) action satisfies two conditions :

(i) $\forall x \in X, e \cdot x = x$, where $e$ is the neutral element of $G$

(ii) $\forall g, h \in G, \forall x \in X,\ g \cdot (h \cdot x) = (g * h) \cdot x$

From these two conditions, we can deduce — for the action ciphers — a property which is similar to the fundamental property of a group cipher.

**Fundamental property** of an action cipher :

If $y = g \cdot x$ then
$x = g^{-1} \cdot y$ where $g^{-1}$ is the inverse of $g$ in $G$.

Proof:
$g^{-1} \cdot y = g^{-1} \cdot (g \cdot x) \overset{(ii)}{=} (g^{-1} * g) \cdot x = e \cdot x \overset{(i)}{=} x$

■

This fundamental property proves that the key-schedule algorithm for deciphering an E/D similar iterated block cipher containing an action cipher component is the same as if we had a group cipher instead of the action cipher and is therefore independant from the particular action itself.

Any group cipher is an action cipher (the group operation always defines an action of the group on itself : $g \cdot x = g * x$). However, there are action ciphers which are not group ciphers even when $X = G$ (see theorem 4.3) ; our new concept of action ciphers really generalizes the classical concept of group ciphers.

Remember now that a group cipher has the following security feature : given a ciphertext $y$, any plaintext $x$ is equally probable if the key is an unknown one-time key and if all possible keys are equally probable. It means that, for a uniformly random one-time key, group ciphers give perfect secrecy.[1]

**Definition 3.2** *An action cipher is perfect if, for a uniformly random one-time key, it gives perfect secrecy.*

[1] Cf. [5] and [2] p.25

Not all action ciphers are perfect. For example, if we take the action induced by the conjugation when $G = X$ is an Abelian group (remember that all three groups in IDEA are Abelian), we get the trivial action. Indeed,

$$g \cdot x = g * x * g^{-1} = g * g^{-1} * x = x, \forall g, x \in G.$$

Such a cipher is just the identity transformation (any plaintext remains unchanged) which is useless from a cryptological point of view. Only action ciphers which are perfect and therefore at least as secure as group ciphers should be used in cryptography.

In the next section, we will characterize the subset of all perfect action ciphers, when $G = X$ is a finite group.

# 4 $\sigma$-action ciphers

In this section, we suppose that $G = X$ is a finite group for the operation $*$. For any permutation $\sigma \in \mathrm{Sym}(G)$, we define the operation $*_\sigma$ as follows :

$$g *_\sigma x = \sigma^{-1}(g * \sigma(x))$$

**Proposition 4.1** *$g \cdot x = g *_\sigma x$ is an action of $G$ on itself.*

Proof:

(i) $\forall x \in X, e \cdot x = \sigma^{-1}(e * \sigma(x)) = \sigma^{-1}(\sigma(x)) = x$

(ii) $\forall g, h \in G, \forall x \in X,$
$g \cdot (h \cdot x) = \sigma^{-1}(g * \sigma(\sigma^{-1}(h * \sigma(x)))) = \sigma^{-1}(g * h * \sigma(x)) = (g * h) \cdot x$

■

**Definition 4.1** *We call the above action — $g \cdot x = g *_\sigma x$ — a $\sigma$-action and the corresponding action cipher a $\sigma$-action cipher.*

Note that the $Id$-action cipher (where $Id$ is the identity permutation) is exactly the classical group cipher :

$$g *_{Id} x = Id^{-1}(g * Id(x)) = g * x$$

In other words, group ciphers are $\sigma$-action ciphers as well.

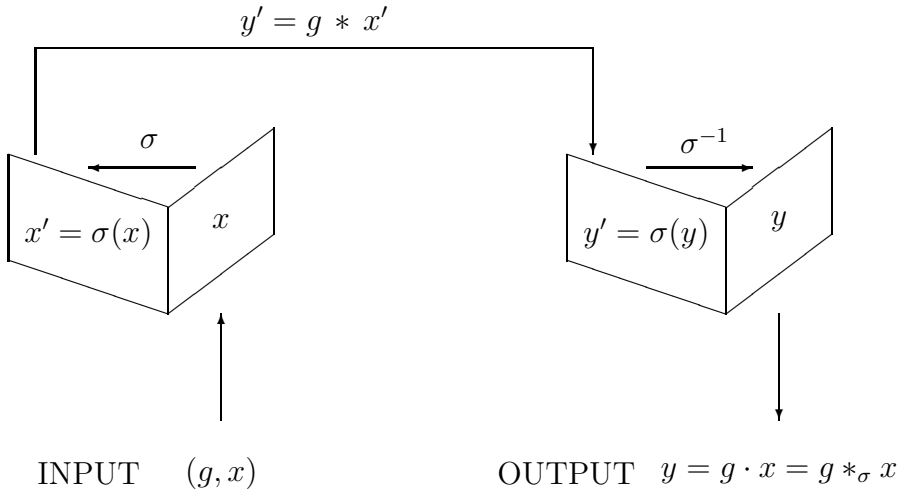
## 4.1 Interpretation

We give here an intuitive interpretation for a $\sigma$-action.

Imagine that the elements of $G$ are kept unchanged while those in $X$ are renamed. The permutation $\sigma$ links the new names with the original ones :

$$\begin{array}{ccc} \{\text{new name (in } X)\} & \longrightarrow & \{\text{original name (in } G)\} \\ x & \overset{\sigma}{\longmapsto} & x' \end{array}$$

Then, the $\sigma$-action can be represented as follows :




To calculate $g \cdot x = g *_\sigma x$, first we find $x'$, the original name for $x$ ; then, we calculate $y' = g * x'$ in the group $G$ and we return $y$, the new name for $y'$.

## 4.2 Properties

First, we want to describe the set of all $\sigma$-actions in more details.

**Definition 4.2** *We say that two permutations $\sigma$ and $\tau$ of $G$ are equivalent relatively to $*$ (or, simply, equivalent) if the corresponding $\sigma$-actions are the same, i.e. if*

$$\forall g, x \in G, \quad g *_\sigma x = g *_\tau x$$

All permutations in a same class of equivalence define the same $\sigma$-action. The following lemma proves that in each class of equivalence there is *at least* one permutation for which $e$, the neutral element of the group, is a fix point.

**Lemma 4.2** *Given a $\sigma$-action, we can suppose, without loss of generality, that $\sigma(e) = e$ where $e$ is the neutral element of the group $G$.*

Proof:
Consider the permutation $\tau$ defined by $\tau(x) = \sigma(x) * \sigma(e)^{-1}$. Then,

$$\tau(e) = \sigma(e) * \sigma(e)^{-1} = e$$

Moreover, $\forall g, x \in G$,

$$\begin{array}{rrcl} & y & = & g *_\sigma x = \sigma^{-1}(g * \sigma(x)) \\ \Longrightarrow & \sigma(y) & = & g * \sigma(x) \\ \Longrightarrow & \sigma(y) * \sigma(e)^{-1} & = & g * \sigma(x) * \sigma(e)^{-1} \\ \Longrightarrow & \tau(y) & = & g * \tau(x) \\ \Longrightarrow & y & = & \tau^{-1}(g * \tau(x)) = g *_\tau x \end{array}$$

This proves that $\forall g, x \in G, g *_\sigma x = g *_\tau x$, i.e. those actions are actually equal.

■

The following lemma proves that in each class of equivalence there is *at most* one permutation for which the neutral element of the group is a fix point.

**Lemma 4.3** *Let be $\sigma$ and $\tau$ two permutations of $G$ such that $\sigma(e) = \tau(e) = e$. If they are equivalent, then they are equal.*

Proof:

$$\begin{array}{lll} & \forall g, x \in G, & \sigma^{-1}(g * \sigma(x)) = \tau^{-1}(g * \tau(x)) \\ \text{for } x = e, & \forall g \in G & \sigma^{-1}(g * \sigma(e)) = \tau^{-1}(g * \tau(e)) \\ \text{as } \sigma(e) = \tau(e) = e, & \forall g \in G & \sigma^{-1}(g) = \tau^{-1}(g) \\ & \Longrightarrow & \sigma = \tau \end{array}$$

■

Corollary: (lemma 4.2 and 4.3)

1) There are exactly $(|G| - 1)!$ different $\sigma$-actions.

2) A permutation $\sigma$ is equivalent to $Id$ — the identity permutation — if and only if $\forall x \in G, \sigma(x) = x * \sigma(e)$.

We want to prove now two important cryptographic properties satisfied by all $\sigma$-action ciphers.

**Proposition 4.4** *For a $\sigma$-action cipher, two different keys always act (i.e. encrypt) differently.*

**Proposition 4.5** *$\sigma$-action ciphers are perfect ; they provide perfect secrecy for uniformly random one-time keys.*

In order to prove those propositions, we need two technical lemmas :

**Lemma 4.6** *All $\sigma$-actions are transitive.*

Proof:
Indeed, $\forall x, y \in G, \exists g \in G$ such that $g \cdot x = y$ : take $g = \sigma(y) * \sigma(x)^{-1}$.

$$\begin{aligned} g \cdot x &= \sigma^{-1}(g * \sigma(x)) \\ &= \sigma^{-1}(\sigma(y) * \sigma(x)^{-1} * \sigma(x)) \\ &= \sigma^{-1}(\sigma(y)) \\ &= y \end{aligned}$$

■

**Lemma 4.7** *Let be $G, *$ a finite group and $\alpha$ an action of $G$ on itself. If $\alpha$ is transitive, then for all $x$ and $y$ in $G$, there is a unique $g$ in $G$ such that $g \cdot x = y$.*

Proof:
The existence of $g$ comes from the definition of a transitive action. We only need to show the uniqueness.

For all $x$ in $G$ we define $\alpha_x : G \longrightarrow G$, $\alpha_x(g) = g \cdot x$. The function $\alpha_x$ is surjective since the action $\alpha$ is transitive ; it is therefore injective. This terminates the proof.

■

Corollary: A $\sigma$-action is "invertible" both on the left and on the right. Indeed, lemmas 4.6 and 4.7 show that all $\sigma$-actions are not only faithful,[2] they even satisfy a simplification rule from the right. Moreover, an action always satisfies a simplification rule from the left :

$$g \cdot x_1 = g \cdot x_2 \Longrightarrow g^{-1} \cdot (g \cdot x_1) = g^{-1} \cdot (g \cdot x_2) \Longrightarrow x_1 = x_2$$

[2] An action is *faithful* when $g \cdot x = h \cdot x, \forall x$ implies $g = h$. In general, $g \cdot x = h \cdot x$ does not imply $g = h$.

Propositions 4.4 and 4.5 are a direct consequence of this corollary.

We prove now the main theorem which characterizes the perfect action ciphers when $G = X$ is a finite group.

**Theorem 4.1** *Let be $G, *$ a finite group and an action of $G$ on itself. The corresponding action cipher is perfect if and only if it is a $\sigma$-action cipher.*

Proof:
Proposition 4.5 proves the theorem in one direction. We only need to prove that any perfect action cipher is actually a $\sigma$-action cipher.

Let's consider a perfect action cipher ; the action must be transitive as any ciphertext might be the encryption of any plaintext. For each $x$ in $G$ there is a unique $g_x$ in $G$ such that $x = g_x \cdot e$ (lemma 4.7). Our candidate for the permutation $\sigma$ is defined by $\sigma(x) = g_x$.

$$\begin{aligned} y &= g *_\sigma x \\ &= \sigma^{-1}(g * \sigma(x)) \\ &= \sigma^{-1}(g * g_x) \\ \sigma(y) &= g * g_x \\ g_y &= g * g_x \\ g_y \cdot e &= (g * g_x) \cdot e \\ g_y \cdot e &= g \cdot (g_x \cdot e) \\ y &= g \cdot x \end{aligned}$$

This proves that our candidate for the $\sigma$-action is indeed equal to the original action.

■

As $G = X$, both $g *_\sigma x$ and $x *_\sigma g$ are defined. Is the operation $*_\sigma$ commutative in general? Is it associative? A good cryptographic operation should not satisfy unneeded algebraic properties. Indeed, unnecessary properties give extra tools to the enemy cryptanalyst.

The following theorem proves that all $\sigma$-actions where $\sigma$ is not equivalent to the identity permutation get rid of two unnecessary properties : commutativity and associativity.

**Theorem 4.2** *When $\sigma$ is not equivalent to $Id$, the $\sigma$-action operation is neither commutative, nor associative.*

Proof:
We show that if the $\sigma$-action operation is commutative (resp. associative), then $\sigma$ is equivalent to $Id$.

1) We suppose that the $\sigma$-action operation is commutative. Then,

$$\begin{array}{lrcl} \forall x \in G, & e \cdot_\sigma x & = & x \cdot_\sigma e \\ \Longrightarrow & \sigma^{-1}(e * \sigma(x)) & = & \sigma^{-1}(x * \sigma(e)) \\ \Longrightarrow & \sigma(x) & = & x * \sigma(e) \end{array}$$

This means that $\sigma$ is equivalent to $Id$ (corollary of lemma 4.2 and 4.3.)

2) We suppose that the $\sigma$-action operation is associative. Then,

$$(a \cdot b) \cdot c \overset{associativity}{=} a \cdot (b \cdot c) \overset{action}{=} (a * b) \cdot c.$$

Therefore, $\forall a, b, c \in G$, we have $(a \cdot b) \cdot c = (a * b) \cdot c$. As a $\sigma$-action is faithful, the above condition means that $a *_\sigma b = a \cdot b = a * b = a *_{Id} b, \forall a, b \in G$, i.e. $\sigma$ is equivalent to $Id$.

■

We can now prove the following theorem that characterizes which $\sigma$-action ciphers are group ciphers as well.

**Theorem 4.3** *A $\sigma$-action cipher is a group cipher if and only if $\sigma$ is equivalent to Id.*

Proof:
"$\Longleftarrow$" The $Id$-action cipher is the usual group cipher.

"$\Longrightarrow$" We suppose that we have a $\sigma$-action cipher which is a group cipher. Then, the action operation is associative. We conclude by applying the theorem 4.2.

■

Theorems 4.2 and 4.3 prove that only a few $\sigma$-action ciphers are actually group ciphers, namely those for which $\sigma$ is equivalent to $Id$ ; most of the $\sigma$-action ciphers are not group ciphers. The $\sigma$-action ciphers create therefore a large family of new operations which are not commutative, not even associative. From a cryptological point of view, this important security feature is a significant improvement in comparison to group ciphers.

Some cryptographic algorithms use two or more different group operations defined on the same set. For example, the three different group operations of IDEA are defined on $\{0, 1, 2, \ldots, 2^{16} - 1\}$.

What happens when we replace two different group ciphers by two $\sigma$-action ciphers? We want to be sure that if we start with two different group ciphers, we always get two *different* $\sigma$-action ciphers after the replacement. In other words, we need to prove that such a replacement can not lead to a unique action cipher.

Theorem 4.4 proves that no collision are possible when we replace group ciphers by $\sigma$-action ciphers. It garanties that two different group operations on the same set cannot lead to the same action operation for particular, well *poorly* chosen permutations.

**Theorem 4.4** *Let be $*$ and $\perp$ two different group operations defined on the same set $G$. Then for all permutations $\sigma$ and $\tau$ of $G$, the actions $*_\sigma$ and $\perp_\tau$ are different.*

Proof:
We will prove that if there are $\sigma$ and $\tau$ two permutations of $G$ such that $*_\sigma$ and $\perp_\tau$ are the same, then $*$ and $\perp$ are already the same.

Let's suppose that for two well chosen permutations $\sigma$ and $\tau$, the actions $*_\sigma$ and $\perp_\tau$ are the same. Let be $e_*$ (resp. $e_\perp$) the neutral element of $G, *$ (resp. $G, \perp$) ; withour loss of generality, we can suppose that $\sigma(e_*) = e_*$ and $\tau(e_\perp) = e_\perp$. As $*_\sigma$ and $\perp_\tau$ are the same, we have

$$(\dagger) \qquad \forall a, b \in G, \qquad \sigma^{-1}(a * \sigma(b)) = \tau^{-1}(a \perp \tau(b)).$$

This is in particular true for $a = e_*$ and $b = e_\perp$ ; the equation $(\dagger)$ becomes

$$\begin{aligned} \sigma^{-1}(e_* * \sigma(e_\perp)) &= \tau^{-1}(e_* \perp e_\perp) \\ e_\perp &= \tau^{-1}(e_*) \end{aligned}$$

In other words, $e_* = \tau(e_\perp) = e_\perp$ ; both groups have the same neutral element $e$, and $\sigma(e) = \tau(e) = e$.

If we replace $b$ by $e$ in equation $(\dagger)$, we see that $\forall a \in G$, $\sigma^{-1}(a) = \tau^{-1}(a)$ ; therefore $\sigma = \tau$.

Then, the equation $(\dagger)$ becomes

$$\begin{aligned} & \sigma^{-1}(a * \sigma(b)) = \sigma^{-1}(a \perp \sigma(b)) && \forall a, b \in G \\ \Longrightarrow \quad & a * \sigma(b) = a \perp \sigma(b) && \forall a, b \in G \\ \Longrightarrow \quad & a * b' = a \perp b' && \forall a, b' \in G \end{aligned}$$

This proves that $*$ is indeed exactly the same group operation as $\perp$.

■

# 5 Conclusion

Most practical block ciphers are E/D similar. Amongst the three classical core components (involutary permutations, involution ciphers and group ciphers) of E/D similar block ciphers, the group ciphers component is with no doubt the most elaborate one. Group ciphers provide perfect secrecy for uniformly random one-time keys. However,

group ciphers are rich structures with unneeded extra properties that can help the enemy cryptanalyst.

Action ciphers generalize the concept of group ciphers. When the group $G$ (the set of all possible keys) and $X$ (the set of all possible plaintexts as well as the set of all possible ciphertexts) are the same, we have characterized the action ciphers that provide perfect secrecy for uniformly random one-time keys : those are exactly the so-called $\sigma$-action ciphers, a subset of action ciphers defined using permutations $\sigma \in \mathrm{Sym}(G)$.

The $\sigma$-action ciphers extend the concept of group ciphers and keep all the good cryptological properties of the group ciphers ; in particular, they provide perfect secrecy for uniformly random one-time keys.

All group ciphers are $\sigma$-action ciphers where the permutation $\sigma$ is equivalent to the identity permutation. But group ciphers are the only $\sigma$-action ciphers for which the ciphering operation is associative. Indeed, when the permutation $\sigma$ is not equivalent to the identity permutation, the $\sigma$-action cipher is not a group cipher and the ciphering operation is neither associative nor commutative, even if it comes from an Abelian group. From a cryptological point of view, this important security feature is a significant improvement in comparison to group ciphers.

Any group cipher component of an E/D similar iterated block cipher can be replaced by a $\sigma$-action cipher. Moreover such a substitution keeps the original key-schedule algorithm.

# References


[1] **J. Calais** *Eléments de théorie des groupes*, puf, 1984.

[2] **X. Lai** *On the Design and Security of Block Ciphers*, Ph.D. thesis, ETH, Zürich, 1992.

[3] **X. Lai and J. L. Massey** *A proposal for a new block encryption standard*, In I.B. Damgård, editor, Advances in Cryptology, Proc. Eurocrypt'90, LNCS 473, p. 389-404, Springer Verlag, 1991.

[4] **W. Meier** *On the security of the IDEA block cipher*, In T. Helleseth, editor, Advances in Cryptology, Proc. Eurocrypt'93, LNCS 765, p. 371-385, Springer Verlag, 1993.

[5] **C. Shannon** *Communication theory of secrecy systems*, Bell system Technical Journal, Vol. 28 p. 656-715, 1949.

# Appendix

## A $\sigma$-action ciphers applied to IDEA

### A.1 Introduction

IDEA is an E/D similar iterated block cipher developped by X. Lai and J. L. Massey at the beginning of the 90s. IDEA has eight rounds in its full version. Each round starts with four group cipher components, followed by an involution cipher component and then by an involutary permutation. We refer the reader to [2] or [3] to find a complete description of this algorithm.

An important design criteria in the developpement of IDEA is the use of so-called "incompatible" group operations which are easy to calculate for a computer. The three group operations are defined on the same set $\{0, 1, 2, \ldots, 2^{16} - 1\}$. The first one, $+$, is the usual addition modulo $2^{16}$ ; the second one, $\odot$, is the multiplication in $\mathbb{F}^*_{2^{16}+1}$ where $2^{16}$ is named 0 ; the third one, $\oplus$, is the "exclusive or" applied bit per bit to the 16-bit words.

The group cipher components only use the $+$ and the $\odot$, as group operations. The involution cipher component contains the $+$ and the $\odot$ operations inside the so-called MA-box[3] as well as the $\oplus$ operation outside of the MA-box.

### A.2 Three "incompatible" group operations

The group operations used in IDEA are incompatible with each other : there is no global distributive law between them. This is an important design criteria.

Indeed, in general,

$$a * (b \perp c) \neq (a * b) \perp (a * c)$$

where $*$ and $\perp$ are any two distinct group operations appearing in IDEA.

Nevertheless, the group operations used in IDEA satisfy some partial distributive laws ([4]) that have been used to attack IDEA up to three rounds. Even though there has been no evidence that this could pose a practical threat for IDEA with eight rounds, the destruction of these partial distributive laws would increase the internal security of the core operations. In the following, we explain how to use $\sigma$-action operations in order to destroy these partial distributive laws inherited from the ring of integers.

[3] "MA" stands for "multiply and add".

## A.3 Replacement of some group operations by $\sigma$-action operations

As already explained, IDEA is an E/D similar iterated block cipher whose rounds contain all three classical E/D similar core transformations : group ciphers, an involution cipher and an involutary permutation.

Only two of the group operations, namely $+$ and $\odot$, intervene in the group cipher components. However, those operations do not only appear in the group cipher components, they are also used internally in the involution cipher component, more precisely in the MA-box. The $\oplus$ group operation only appears outside the MA-box in the involution cipher component. The involutary permutation does not contain any group operation.

Replacing some of the group operations in IDEA by $\sigma$-action operations can indeed destroy the partial distributive laws inherited from the ring of integers and consequently make the core operations in IDEA even more incompatible with each other.

Moreover, the new operations are neither commutative nor associative if the permutation is not equivalent to the identity permutation. This is an improvement from a cryptological point of view : this destroys unnecessary algebraic properties that can help the enemy cryptanalyst. Because of the non commutativity, the order of the operands is important.

### A.3.1 Group operations appearing in the group cipher components

We choose $\sigma$ and $\tau$, two permutations of $\{0, 1, 2, \ldots, 2^{16} - 1\}$. Then, we systematically substitute $+$ (resp. $\odot$) by $+_\sigma$ (resp. $\odot_\tau$) in the group cipher components.

If the permutations $\sigma$ and $\tau$ are not equivalent to the identity permutation, the operations $+_\sigma$ and $\odot_\tau$ are neither commutative, nor associative. Moreover, theorem 4.4 garanties that $+_\sigma$ and $\odot_\tau$ are always different from each other, for any choice of the permutations $\sigma$ and $\tau$.

As we know, replacing $+$ (resp. $\odot$) by $+_\sigma$ (resp. $\odot_\tau$) in the group cipher components keeps the original key-schedule algorithm.

Using $\sigma$-action ciphers to replace the group ciphers in IDEA has another consequence that goes beyond the original scope of E/D similar core components. Introducing $\sigma$-action ciphers components creates a large family of IDEA-like algorithms which can be customized. All these algorithms look like the original IDEA and have exactly the same key-schedule algorithm as IDEA.

### A.3.2 Group operations appearing in the involution cipher component

In the involution, the operations + and $\odot$ only appear in a so-called MA-box. With Lai's notation, the MA-box transformation can be seen as a function $(V_1, V_2) = MA(U_1, U_2, Z_5, Z_6)$ which has the following important properties (Cf. [2]) :

- *for any choice of the key subblocks $Z_5$ and $Z_6$, $MA(\cdot, \cdot, Z_5, Z_6)$ is an invertible transformation ; for any choice of $U_1$ and $U_2$, $MA(U_1, U_2, \cdot, \cdot)$ is also an invertible transformation ;*
- *this structure has a "complete diffusion" effect in the sense that each output subblock depends on every input subblock...*

If we replace + by $+_\sigma$ and $\odot$ by $\odot_\tau$ these properties stay valid.[4] The proof only uses the fact that these operations are "invertible" on both sides, which is the case for any $\sigma$-action operation (see corollary on page 14). Such a replacement is therefore totally compatible with the MA-box design and structure. Eventually, it has no impact on the key-schedule algorithm and, moreover, the whole involution component stays involutive.

Two approaches are possible : either we replace + and $\odot$ by $\sigma$-action operations only in the group cipher components and we keep unchanged these operations in the involution component, or we systematically substitute all the group operations + (resp. $\odot$) by $+_\sigma$ (resp. $\odot_\tau$) in all the components.

The involution cipher component contains also the third group operation, namely $\oplus$.

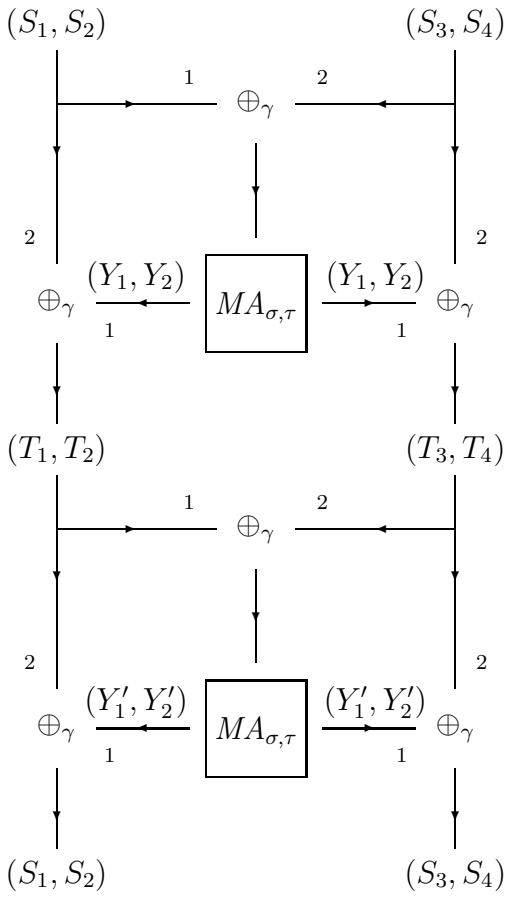
Let's consider a permutation $\gamma$ of $\{0, 1, 2, \ldots, 2^{16} - 1\}$ and the corresponding $\sigma$-action operation $\oplus_\gamma$. As we will see, most of the permutations $\gamma$ destroy the involutive property of the involution component. Indeed :

**Proposition A.1** *The operation $\oplus_\gamma$ must be equal to $\oplus$ in order for the involution component in IDEA to stay involutive.*

Proof:
Let's consider two iterations of the whole involution component where $\oplus$ has been replaced by $\oplus_\gamma$.

As $\oplus_\gamma$ is not commutative in general, the order of the operands is important again. In the following diagram, the order of the operands is indicated by "1" and "2", representing respectively the left and the right operands.

[4] As the new operations can be non commutative, the order of the operands has to be fixed.




We observe that $T_1 = Y_1 \oplus_\gamma S_1$ and $S_1 = Y_1' \oplus_\gamma T_1$.

It means that

$$0 \oplus_\gamma T_1 \overset{action}{=} T_1 = Y_1 \oplus_\gamma S_1 = Y_1 \oplus_\gamma (Y_1' \oplus_\gamma T_1) \overset{action}{=} (Y_1 \oplus\ Y_1') \oplus_\gamma T_1.$$

By the corollary on page 14, we conclude that $Y_1 \oplus\ Y_1' = 0$, i.e. $Y_1 = Y_1'$. Similarly, $Y_2 = Y_2'$.

As MA-boxes are invertible for a fixed choice of the keys, the equalities $Y_i = Y_i', i = 1, 2$ imply that the input of the MA-boxes is the same on both levels. In other words, we need to have $S_1 \oplus_\gamma S_3 = T_1 \oplus_\gamma T_3$ and $S_2 \oplus_\gamma S_4 = T_2 \oplus_\gamma T_4$.

This means :

$$\begin{aligned} S_1 \oplus_\gamma S_3 &= (Y_1 \oplus_\gamma S_1) \oplus_\gamma (Y_1 \oplus_\gamma S_3) \qquad \text{and} \\ S_2 \oplus_\gamma S_4 &= (Y_2 \oplus_\gamma S_2) \oplus_\gamma (Y_2 \oplus_\gamma S_4) \end{aligned}$$

The operation $\oplus_\gamma$ must therefore satisfy :

$$\forall Y, S, S' \in G, \qquad S \oplus_\gamma S' = (Y \oplus_\gamma S) \oplus_\gamma (Y \oplus_\gamma S').$$

Without loss of generality (lemma 4.2), we can suppose $\gamma(0) = 0$. For $S' = 0$ the above relation becomes :

$$\begin{array}{rrcll}
 & \gamma^{-1}(S \oplus \gamma(0)) & = & \gamma^{-1}((Y \oplus_\gamma S) \oplus \gamma(Y \oplus_\gamma 0)) & \forall Y, S \in G \\
 & & = & \gamma^{-1}((Y \oplus_\gamma S) \oplus \gamma(\gamma^{-1}(Y \oplus \gamma(0)))) & \forall Y, S \in G \\
 & \gamma^{-1}(S) & = & \gamma^{-1}((Y \oplus_\gamma S) \oplus\ Y) & \\
\Longrightarrow & S & = & (Y \oplus_\gamma S) \oplus\ Y & \forall Y, S \in G \\
\Longrightarrow & Y \oplus_\gamma S & = & Y \oplus\ S & \forall Y, S \in G
\end{array}$$

In other words, both operations $\oplus_\gamma$ and $\oplus$ must be equal.

■

In conclusion, if we want to keep only three operations in our generalizations of IDEA, the group operation $+$ (resp. $\odot$) can be replaced by any $\sigma$-action operation $+_\sigma$ (resp. $\odot_\tau$), but the group operation $\oplus$ has to be kept unchanged. The theorem 4.4 garanties that for any choice of the permutations $\sigma$ and $\tau$, the operation $+_\sigma$, $\odot_\tau$ and $\oplus$ will always be different from each other.

## A.4 How to choose $+_\sigma$ and $\odot_\tau$?

Any permutation which is not equivalent to the identity permutation leads to a ciphering operation which is neither associative, nor commutative. Moreover, it is very likely to destroy most of the (partial) algebraic properties of the group operations, properties which are inherited from the ring of integers.[5] Using $\sigma$-action operations can help make the core operations even more resistant to linear and differential cryptanalysis.

The core ciphering operations must be easy to handle for a computer both in hardware and software. This is an important criteria in the choice of the group operations of IDEA.

To define $+_\sigma$ and $\oplus_\tau$ we need two permutations of $\{0, 1, 2, \ldots, 2^{16} - 1\}$.

We can choose for $\sigma$ and $\tau$ any permutations of $\{0, 1, 2, \ldots, 2^{16} - 1\}$ if enough memory is available. For the choice of $+_\sigma$ and $\odot_\tau$, the total entropy is greater than $1'908'040$. From a practical point of view, if we want to store $\sigma$, $\sigma^{-1}$, $\tau$ and $\tau^{-1}$ in lookup tables, we need half a megabyte...

However, we can imagine cheaper and faster ways to define (less general) permutations of $\{0, 1, 2, \ldots, 2^{16} - 1\}$. If speed is important, we can simply swap the halves for example, i.e. $\sigma(x) = \sigma(x_L|x_R) = x_R|x_L$, where the vertical bar $|$ represents the concatenation, $x_L$ is the left half of $x$ (8 most significant bits) and $x_R$ is the right half of $x$ (8 least significant bits.)

[5] Notice, for example, that if we consider only the least significant bit, both operations $+$ and $\oplus$ are the same.

A trade off between speed and entropy is to "split" the permutations in two parts :

$$\sigma(x) = \sigma(x_L|x_R) = \sigma_L(x_L)|\sigma_R(x_R)$$

where $\sigma_L$ and $\sigma_R$ are two permutations of $\{0, 1, 2, \ldots, 255\}$. Since we can fix the image of the neutral element, we can actually choose $\sigma_L$ and $\sigma_R$ among $(255!)^2$ different $\sigma$-actions. If we split both $\sigma$ and $\tau$, the total entropy for the choice of $+_\sigma$ and $\odot_\tau$ is slightly greater than 6700 ; it requires only 2 Kbytes of memory to store $\sigma$, $\sigma^{-1}$, $\tau$ and $\tau^{-1}$.

## A.5 Conclusion

We have shown how some of the group operations appearing in IDEA can be replaced by $\sigma$-action operations without modifying the original key-schedule algorithm of IDEA.

The $\oplus$ group operation have to be kept in order to maintain the involutive property of the involution component. The other group operations, $+$ and $\odot$, can be replaced by $\sigma$-action operations. This replacement can occur either in the group cipher components only or systematically in all the components. Those new operations are neither commutative nor associative when the permutations are not equivalent to the identity permutation.

Using $\sigma$-action operations to replace some of the group oprations in IDEA has a consequence that goes beyond the original scope of core components in E/D similar iterated block ciphers. Introducing $\sigma$-action ciphers components creates a large family of IDEA-like algorithms which can be customized. If we call IDEAS (International Data Encryption Algorithms Systems) this large family of IDEA-like ciphers, all algorithms in IDEAS look like the original IDEA and have exactly the same key-schedule algorithm as IDEA. This process of customization allows the user to take advantage of a public, well studied algorithm while keeping his/her own version secret.

Using $\sigma$-action operations can destroy the partial distributive laws of $+$, $\odot$ and $\oplus$ in IDEA, inherited from the ring of integers. This can make the core operations even more resistant to linear and differential cryptanalysis. This is a significant improvement from a cryptological point of view : it destroys unnecessary algebraic properties in IDEA that help the enemy cryptanalyst.