# Chapter 13. Schreier-Sims Method

Finally, we explain how to construct a base and strong generating set, given a set of generators. The method is founded on a result by Schreier, and was first developed by Sims. After presenting the original Schreier-Sims method, this chapter will discuss some variations.

## Verifying Strong Generation

The "classical" viewpoint for the Schreier-Sims method is that we have a (partial) base $B$ and a set $S$ of elements of a group $G$, where $S$ contains a generating set of $G$. We wish to verify that $B$ is indeed a base, and that $S$ is a strong generating set relative to $B$. If we discover that this is not so, then we wish to extend either $B$ or $S$, or both, until it is true.

The simplest case is where $S$ is the original set of generators and we choose some points $[\beta_1, \beta_2, ..., \beta_k]$ to form $B$, so that no generator fixes all $k$ points.

Let

$$
\begin{aligned}
S^{(i)} &= \{ \ s \in S \ \mid \ s \text{ fixes } \beta_1, \beta_2, ..., \beta_{i-1} \ \}, \\
H^{(i)} &= < S^{(i)} >, \ \text{ and} \\
G^{(i)} &= G_{\beta_1, \beta_2, ..., \beta_{i-1}}, \ \ 1 \le i \le k+1.
\end{aligned}
$$

Hence, $H^{(k+1)} = \{id\}$. To verify that $B$ is a base and $S$ is a strong generating set, we need to show that

$$H^{(i)} = G^{(i)}, \ \text{ for all } i, \ \ 1 \le i \le k+1.$$

Once again, we will use an inductive approach, working from the bottom of the base to the top. In this way, we have a nice inductive hypothesis :

### Hypothesis

Assume that $B$ is a base for $H^{(i+1)}$ and that $S^{(i+1)}$ is a strong generating set of $H^{(i+1)}$ relative to $B$.

To prove the inductive hypothesis for $i$ from the hypothesis for $i+1$, we need to show that

$$H^{(i)}{}_{\beta_i} = H^{(i+1)}.$$

Furthermore, we know that $H^{(1)} = G^{(1)} = G$. So, if we have proved that $B$ is a base of $H^{(1)}$ and that $S = S^{(1)}$ is a strong generating set of $H^{(1)}$ relative to $B$, then we have the same result for $G$.

Not only does the inductive approach provide a neat proof of correctness, but it also allows us to assume we have a base and strong generating set of $H^{(i+1)}$. This allows us to easily answer questions involving $H^{(i+1)}$ and elements of $G$, such as membership.

An outline of an algorithm based on the above inductive hypothesis is presented as Algorithm 1.

**Algorithm 1 : Outline of Schreier-Sims method**

```
Input : a set S of generators of a group G;

Output : a base B for G;
        a strong generating set S of G relative to B;

procedure Schreier−Sims( var B : partial base; var S : set of elements; i : integer );
(* Assuming that B and S^(i+1) are a base and strong generating set
   for H^(i+1), produce a base and strong generating set for H^(i). *)
begin

  while H^(i)_βi ≠ H^(i+1) do

    find g ∈ H^(i)_βi − H^(i+1); find largest j such that g fixes β1, β2, ..., βj−1;

    add g to S;   (*actually to S^(i+1), S^(i+2), ...,S^(j) *)

    (*extend base, if necessary, so that no strong generator fixes all the base points*)
    if j = k+1 then
      find βj not fixed by g;
      add βj to B;
    end if;

    (*ensure we still have a base and strong generating set for H^(i+1) *)
    for level := j downto i+1 do
      Schreier−Sims( B, S, level );
    end for;
  end while;
end;

begin
  find points β1, β2, ..., βk so that no element of S fixes all of them;
  B := [β1, β2, ..., βk];
  for i := k downto 1 do
    Schreier−Sims( B, S, i );
  end for;
end;
```

The element $g \in H^{(i)}{}_{\beta_i} - H^{(i+1)}$ found in procedure *Schreier–Sims* will alter $S^{(j)}$, $S^{(j-1)}$, ..., $S^{(i+1)}$, and hence our assumptions about $H^{(j)}$, $H^{(j-1)}$, ..., $H^{(i+1)}$. Furthermore, if $g$ fixes all the points presently in $B$ then $B$ must be extended.

## Schreier Generators

The two open questions for the completion of the outline of the Schreier-Sims method are

1. How do we test $H^{(i)}{}_{\beta_i} = H^{(i+1)}$, and
2. How do we find an element $g \in H^{(i)}{}_{\beta_i} - H^{(i+1)}$?

The answer lies in the following result, which we will not prove. It is similar to the Loop Basis Theorem of Chapter 5.

### Schreier's Lemma

Let $v^{(i)}$ be a Schreier vector of $\beta_i$ under $H^{(i)}$ relative to the set $S^{(i)}$ of generators of $H^{(i)}$. Then $H^{(i)}{}_{\beta_i}$ is generated by

$$\{ \ trace(\gamma, v^{(i)}) \times s \times trace(\gamma^s, v^{(i)})^{-1} \ \mid \ \gamma \in \beta_i^{H^{(i)}}, s \in S^{(i)} \ \}.$$

The members of the above generating set are called *Schreier generators*.

The answer to both questions is to run through the Schreier generators - all

$$|\beta_i^{H^{(i)}}| \times |S^{(i)}|$$

of them - and test if they are in $H^{(i+1)}$. The membership test is straightforward because we have a base and strong generating set of $H^{(i+1)}$. If all the Schreier generators are in $H^{(i+1)}$, then $H^{(i)}{}_{\beta_i} = H^{(i+1)}$. If not, then any Schreier generator that is not in $H^{(i+1)}$ provides an element $g \in H^{(i)}{}_{\beta_i} - H^{(i+1)}$.

The fleshed out algorithm is presented as Algorithm 2.

**Algorithm 2 : Using Schreier Generators in Schreier-Sims method**

Input : a set $S$ of generators of a group $G$;

Output : a base $B$ for $G$;
a strong generating set $S$ of $G$ relative to $B$;

```
procedure Schreier-Sims( var B : partial base; var S : set of elements; i : integer );
(* Assuming that B and S^(i+1) are a base and strong generating set
   for H^(i+1), produce a base and strong generating set for H^(i).
   Note that H^(i) is invariant during the execution, and that
   the initial S^(i) is a set of generators of H^(i). *)
begin
  gen_set := S^(i);
  for each γ ∈ Δ^(i) do
    for each generator s ∈ gen_set do

      g := trace( γ, v^(i) ) × s × trace( γ^s, v^(i) )^-1;

      if g ∉ H^(i+1) then
        find largest j such that g fixes β_1, β_2, ..., β_(j-1);
        add g to S;  (*actually to S^(i+1), S^(i+2), ...,S^(j) *)
        (*extend base, if necessary, so that no strong generator
         fixes all the base points*)
        if j = k+1 then
          find β_j not fixed by g;
          add β_j to B;
        end if;

        (*ensure we still have a base and strong generating set for H^(i+1) *)
        for level := j downto i+1 do
          Schreier-Sims( B, S, level );
        end for;
      end if;

    end for;
  end for;
end;

begin
  find points β_1, β_2, ..., β_k so that no element of S fixes all of them;
  B := [β_1, β_2, ..., β_k];
  for i := k downto 1 do
    Schreier-Sims( B, S, i );
  end for;
end;
```

## Example

We will execute Algorithm 2 using the symmetries of the projective plane of order two. The group is generated by $a$=(1,2,4,5,7,3,6), and $b$=(2,4)(3,5). We initially take $S=\{a, b\}$. We choose the initial partial base to be $B$ = [1,2].

The first call to the procedure *Schreier–Sims* from the main algorithm is

$$Schreier\text{–}Sims\,(\,[1,2],\ \{a,b\},\ 2\,).$$

The relevant Schreier vector for forming the Schreier generators is

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $v^{(2)}$ | 0 | 0 | 0 | $b$ | 0 | 0 | 0 |

The Schreier generators considered are

$id \times b \times b^{-1} = id$, for $\gamma = 2$, and

$b \times b \times id^{-1} = id$, for $\gamma = 4$.
Both are in $H^{(3)} = \{id\}$.

The next call to *Schreier–Sims* from the main algorithm is

$$Schreier\text{–}Sims\,(\,[1,2],\ \{a,b\},\ 1\,).$$

The relevant Schreier vector for forming the Schreier generators is

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $v^{(1)}$ | 0 | $a$ | $a$ | $a$ | $a$ | $a$ | $a$ |

The Schreier generators considered are

$id \times a \times a^{-1} = id$, for $\gamma = 1$;

$id \times b \times id^{-1} = b \in H^{(2)}$, for $\gamma = 1$;

$a \times a \times a^{-2} = id$, for $\gamma = 2$;

$a \times b \times a^{-2}$ = (2,6,3,7)(4,5) = $g_1$, for $\gamma$ = 2; This is added to $S$, however, the base is not extended.

There is now a call to *Schreier–Sims* with $i$ = 2 from the body of the procedure with $i$ = 1. The call is

$$Schreier\text{–}Sims\,(\,[1,2],\ \{a,b,g_1\},\ 2\,).$$

The relevant Schreier vector for forming the Schreier generators is

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $v^{(2)}$ | 0 | 0 | $g_1$ | $b$ | $g_1$ | $g_1$ | $g_1$ |

The Schreier generators considered are

$id \times b \times b^{-1} = id$, for $\gamma = 2$;

$id \times g_1 \times g_1^{-1} = id$, for $\gamma = 2$;

$g_1^2 \times b \times (b \times g_1)^{-1} = (4,7)(5,6) = g_2$, for $\gamma = 3$; This is added to $S$, and the base is extended by $\beta_3 = 4$. The call

$$Schreier\text{--}Sims\,(\,[1,2,4],\ \{a,b,g_1,g_2\},\ 3\,).$$

verifies that we have a base and strong generating set for $H^{(3)} = \langle g_2 \rangle$.
Back at $i = 2$, the processing of Schreier generators continues as follows:

$g_1^2 \times g_1 \times g_1^{-3} = id$, for $\gamma = 3$;

$b \times b \times id^{-1} = id$, for $\gamma = 4$;

$b \times g_1 \times (b \times g_1)^{-1} = id$, for $\gamma = 4$;

$(b \times g_1) \times b \times g_1^{-2} = g_2$, for $\gamma = 5$;

$(b \times g_1) \times g_1 \times b^{-1} = (4,5)(6,7) = g_3$, for $\gamma = 5$; This is added to $S$, however, the base is not extended. The call

$$Schreier\text{--}Sims\,(\,[1,2,4],\ \{a,b,g_1,g_2,g_3\},\ 3\,).$$

verifies that we have a base and strong generating set for $H^{(3)} = \langle g_2, g_3 \rangle$.
Back at $i = 2$, the processing of Schreier generators continues as follows:

$g_1 \times b \times g_1^{-1} = g_2 \times g_3$, for $\gamma = 6$;

$g_1 \times g_1 \times g_1^{-2} = id$, for $\gamma = 6$;

$g_1^3 \times b \times g_1^{-3} = g_2$, for $\gamma = 7$;

$g_1^3 \times g_1 \times id^{-1} = id$, for $\gamma = 7$.

Thus producing a base and strong generating set for $H^{(2)}$.

Back at $i = 1$, the processing of Schreier generators continues as follows:

$a^5 \times a \times a^{-6} = id$, for $\gamma = 3$;

$a^5 \times b \times a^{-3} = g_3 \times b \times g_1$, for $\gamma = 3$;

$a^2 \times a \times a^{-3} = id$, for $\gamma = 4$;

$a^2 \times b \times a^{-1} = g_1$, for $\gamma = 4$;

$a^3 \times a \times a^{-4} = id$, for $\gamma = 5$;

$a^3 \times b \times a^{-5} = (g_3 \times b \times g_1)^{-1}$, for $\gamma = 5$;

$a^6 \times a \times id^{-1} = id$, for $\gamma = 6$;

$a^6 \times b \times a^{-6} = g_3$, for $\gamma = 6$;

$a^4 \times a \times a^{-5} = id$, for $\gamma = 7$;

$a^4 \times b \times a^{-5} = g_3 \times g_1$, for $\gamma = 7$.

This completes the construction of a base and strong generating set. The result is a base

$$[1,2,4]$$

and a strong generating set

$a$=(1,2,4,5,7,3,6), $b$=(2,4)(3,5),
$g_1$=(2,6,3,7)(4,5), $g_2$=(4,7)(5,6), and $g_3$=(4,5)(6,7).

Note that the element $g_1$ is redundant as a strong generator. Further note that the second call to the procedure *Schreier–Sims* with $i = 2$ rechecked the Schreier generators corresponding to $\gamma = 2$ and 4 and generator $b$. The second call to the procedure *Schreier–Sims* with $i = 3$ rechecked the Schreier generators corresponding to $\gamma = 4$ and 7 and generator $g_2$.

## Avoid Rechecking Schreier Generators

During the example, Algorithm 2 calls procedure *Schreier–Sims* with $i = 2$ twice. Each time the complete set of Schreier generators is checked for membership in $H^{(3)}$. At the first call

$$H^{(2)} = < b >, \quad H^{(3)} = \{id\}, \quad \Delta^{(2)} = \{2,4\},$$

and the Schreier vector $v^{(2)}$ is

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $v^{(2)}$ | 0 | 0 | 0 | $b$ | 0 | 0 | 0 |

A subscript 1 will distinguish these values. Thus, we will speak of $H^{(2)}{}_1$, $H^{(3)}{}_1$, $\Delta^{(2)}{}_1$, and $v^{(2)}{}_1$.

We can arrange for the Schreier vector $v^{(2)}$ to be extended whenever $H^{(2)}$ is extended. So the second call to the procedure *Schreier–Sims* has

$$H^{(2)} = < b, g_1 >, \quad H^{(3)} = \{id\}, \quad \Delta^{(2)} = \{2,3,4,5,6,7\},$$

and the Schreier vector $v^{(2)}$ is

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $v^{(2)}$ | 0 | 0 | $g_1$ | $b$ | $g_1$ | $g_1$ | $g_1$ |

A subscript 2 will distinguish these values. Thus, we will speak of $H^{(2)}{}_2$, $H^{(3)}{}_2$, $\Delta^{(2)}{}_2$, and $v^{(2)}{}_2$.

The extension of $v^{(2)}{}_1$ to $v^{(2)}{}_2$ is important. It allows us to claim that

$$trace(\gamma, v^{(2)}{}_2) = trace(\gamma, v^{(2)}{}_1), \text{for all } \gamma \in \Delta^{(2)}{}_1$$

and that the Schreier generator

$$\begin{aligned} & trace(\gamma, v^{(2)}{}_2) \times s \times trace(\gamma^s, v^{(2)}{}_2)^{-1} \\ = \; & trace(\gamma, v^{(2)}{}_1) \times s \times trace(\gamma^s, v^{(2)}{}_1)^{-1} \end{aligned}$$

for all $\gamma \in \Delta^{(2)}{}_1$ and all generators $s$ of $H^{(2)}{}_1$. As the first call to *Schreier–Sims* (with $i$=2) has verified that these Schreier generators are in $H^{(3)}$, there is no need for the second call to recheck this fact. Even if $H^{(3)}$ changes value, this is so, because the only possible change to $H^{(3)}$ is for $H^{(3)}$ to be extended.

The argument generalizes to show that, provided the Schreier vectors are calculated by extending their previous value, a call to *Schreier–Sims* with the value $i$ does not need to recheck the Schreier generators considered by the previous calls the *Schreier–Sims* with the same value of $i$.

Algorithm 3 avoids the rechecking of Schreier generators. A further parameter $T$ is introduced for the procedure. The parameter $T$ is the subset of the generators $S^{(i)}$ that lie outside the previous value of $H^{(i)}$. That is, the generators $s$ whose Schreier generators

$$trace(\gamma, v^{(i)}) \times s \times trace(\gamma^s, v^{(i)})^{-1}$$

for $\gamma$ in the previous value of $\Delta^{(i)}$ are not yet known to lie in $H^{(i+1)}$.

**Algorithm 3 : Schreier-Sims Method not rechecking Schreier Generators**

Input : a set $S$ of generators of a group $G$;

Output : a base $B$ for $G$;
a strong generating set $S$ of $G$ relative to $B$;

**procedure** *Schreier–Sims*( **var** $B$ : partial base; **var** $S$ : set of elements;
$i$ : integer; $T$ : set of elements );
(* Assuming that $B$ and $S^{(i+1)}$ are a base and strong generating set
for $H^{(i+1)}$, produce a base and strong generating set for $H^{(i)}$

$T$ is the set of additional generators in $S^{(i)}$ since the previous call
to the procedure with the present value of $i$.

Assume that a base and strong generating set of $<S^{(i)} - T>$,
(the previous value of $H^{(i)}$), are included in $B$ and $S$.

The present value of $v^{(i)}$ must be an extension of the previous value. *)

```
begin
  current_gens := S^(i); old_gens := S^(i) - T;
  old_Δ := β_i^<old_gens>; (*previous value of Δ^(i) *)

  for each γ ∈ Δ^(i) do

    if γ ∈ old_Δ then
      gen_set := T;
    else
      gen_set := current_gens;
    end if;

    for each generator s ∈ gen_set do

      g := trace( γ, ν^(i) ) × s × trace( γ^s, ν^(i) )^-1;

      if g ∉ H^(i+1) then
        find largest j such that g fixes β_1, β_2, ..., β_(j-1);
        add g to S;  (*actually to S^(i+1), S^(i+2), ...,S^(j) *)
        (*extend base, if necessary, so that no strong generator
        fixes all the base points*)
       if j = k+1 then
          find β_j not fixed by g;
          add β_j to B;
        end if;
        (*ensure we still have a base and strong generating set for H^(i+1) *)
        for level := j downto i+1 do
          Schreier-Sims( B, S, level, {g} );
        end for;
      end if;
    end for;

  end for;
end;

begin
  find points β_1, β_2, ..., β_k so that no element of S fixes all of them;
  B := [β_1, β_2, ..., β_k];
  for i := k downto 1 do
    Schreier-Sims( B, S, i, S^(i) );
  end for;
end;
```

## Stripping Schreier Generators

Let us take a closer look at testing $g \in H^{(i+1)}$. The test attempts to express $g$ as

$$g = u_k \times u_{k-1} \times \cdots \times u_{i+1}$$

for suitable $u_j \in H^{(j)}$ determined from the Schreier vectors. If the test fails, it is because some suitable $u_l$, $k \leq l \leq i+1$, cannot be found. Thus

$$g = \bar{g} \times u_{l-1} \times u_{l-2} \times \cdots \times u_{i+1}$$

where $u_j \in H^{(j)}$ and $\bar{g} \notin H^{(l)}$. We call $\bar{g}$ the *residue* of testing $g \in H^{(i+1)}$. If $g \in H^{(i+1)}$ then the residue is the identity. The process of determining the residue is called *stripping*.

When $g$ is added to $S$, and procedure *Schreier–Sims* is called at level $i+1$, it must eventually extend $H^{(l)}$ by some generator related to $\bar{g}$. However, $\bar{g}$ and $g$ are not independent. In fact, $g$ will be a redundant generator of $H^{(i+1)}$ once $\bar{g}$ is added to $S$. So, why not just add $\bar{g}$ to $S$ in the first instance, and forget about adding $g$. This not only leads to smaller strong generating sets, but also extends $H^{(l)}$ much sooner. This idea is used in Algorithm 4.

**Algorithm 4 : Schreier-Sims Method stripping Schreier Generators**

Input : a set $S$ of generators of a group $G$;

Output : a base $B$ for $G$;
    a strong generating set $S$ of $G$ relative to $B$;

**procedure** *Schreier–Sims*( **var** $B$ : partial base; **var** $S$ : set of elements;
    $i$ : integer; $T$ : set of elements );
(* Assuming that $B$ and $S^{(i+1)}$ are a base and strong generating set
  for $H^{(i+1)}$, produce a base and strong generating set for $H^{(i)}$.

  $T$ is the set of additional generators in $S^{(i)}$ since the previous call
  to the procedure with the present value of $i$.

  Assume that a base and strong generating set of $<S^{(i)} - T>$,
  (the previous value of $H^{(i)}$), are included in $B$ and $S$.

  The present value of $\nu^{(i)}$ must be an extension of the previous value. *)

```
begin
  current_gens := S^(i); old_gens := S^(i) − T;
  old_Δ := β_i^<old_gens>; (*previous value of Δ^(i) *)

  for each γ ∈ Δ^(i) do

    if γ ∈ old_Δ then
      gen_set := T;
    else
      gen_set := current_gens;
    end if;

    for each generator s ∈ gen_set do

      g := trace( γ, v^(i) ) × s × trace( γ^s, v^(i) )^−1;

      if g ∉ H^(i+1) then

        ḡ := residue of testing g ∈ H^(i+1);
        j := level l where testing stopped;   (*may be k+1*)

        add ḡ to S;  (*actually to S^(i+1), S^(i+2), ...,S^(j) *)
        (*extend base, if necessary, so that no strong generator
        fixes all the base points*)
       if j = k+1 then
         find β_j not fixed by ḡ;
         add β_j to B;
       end if;
       (*ensure we still have a base and strong generating set for H^(i+1) *)
       for level := j downto i+1 do
         Schreier−Sims( B, S, level, {ḡ} );
       end for;
      end if;
    end for;

  end for;
end;

begin
  find points β_1, β_2, ..., β_k so that no element of S fixes all of them;
  B := [β_1, β_2, ..., β_k];
  for i := k downto 1 do
    Schreier−Sims( B, S, i, S^(i) );
  end for;
end;
```

## Variations of the Schreier-Sims Method

This section will discuss some variations of the Schreier-Sims method. They are all variations on Algorithm 4. They vary sometimes in the amount of information known at the start - for example, a base may be known - but mostly they differ in strategies to save space and time.

**Original Schreier-Sims Method:** The original algorithm that Sims devised used coset representatives rather than Schreier vectors. While being more space-consuming, empirical evidence indicates it is a factor of three faster.

**Random Schreier-Sims Method:** If $H^{(i)}{}_{\beta_i} \neq H^{(i+1)}$ then $H^{(i+1)}$ is a proper subgroup of $H^{(i)}{}_{\beta_i}$. Therefore, it has index at least two. This means that the probability of finding an element $g \in H^{(i)}{}_{\beta_i} - H^{(i+1)}$ is *at least* one half.

The random Schreier-Sims method tests $H^{(i)}{}_{\beta_i} = H^{(i+1)}$ by considering a number of (hopefully) random elements $g$ of $G$ and testing whether $g \in H^{(1)}$. If the residue $\bar{g}$ is not trivial, then $\bar{g}$ is a new strong generator. If $t$ consecutive random elements of $G$ are stripped to the identity then the probability that $B$ and $S$ are a base and strong generating set is $1-2^{-t}$.

**Schreier-Todd-Coxeter-Sims Method:** This method not only constructs a base and strong generating set, but also constructs a set of defining relations for the group $G$ involving all the strong generators. The Todd-Coxeter algorithm can compute the index of $H^{(i+1)}$ in $H^{(i)}$, provided sufficient relations are known. The index should be $|\Delta^{(i)}|$. If there are insufficient relations, or the index is too large, the output of the Todd-Coxeter algorithm indicates which words in the generators $S^{(i)}$ it believes are the coset representatives of $H^{(i+1)}$ in $H^{(i)}$. Checking the image of $\beta_i$ under these words will discover two words $w_1$ and $w_2$ that actually represent the same coset. Let $g = w_1 \times w_2^{-1}$. Then $g \in H^{(i)}{}_{\beta_i}$. Either $g \in H^{(i+1)}$ and we obtain another relation, or $g \notin H^{(i+1)}$ and we obtain a new strong generator. This process iterates until the Todd-Coxeter algorithm does compute the index $|\Delta^{(i)}|$.

**Extending Schreier-Sims Method:** Given a base $B$ and strong generating set $S$ of a group $G$ and an element $g \notin G$, we find a base and strong generating set of $<G, g>$. This is simply a call

$$Schreier\text{-}Sims(\, B,\, S \cup \{g\},\, 1,\, \{g\}\,)$$

to the procedure of Algorithm 4.

This task is frequently used in other algorithms, for example, those algorithms of chapters 4 and 6. In most contexts we are extending a subgroup of a group for which we know a base. This not only gives us a base for the extended subgroup, but also allows the formation of Schreier generators and their stripping to be done in terms of base images. A complete permutation is required only in the few cases which lead to a new strong generator. This variation is called the **known base Schreier-Sims method.**

## Summary

The Schreier-Sims method produces a base and strong generating set of a group given by generators. It does this by verifying that all the Schreier generators can be expressed in terms of coset representatives.

There are several variations on the Schreier-Sims method.

## Exercises

(1/Moderate) The Schreier-Sims methods are very tedious to perform by hand for all but the smallest examples. Execute Algorithm 3 on the symmetries of the square, the symmetric group of degree 4, and the automorphism group of Petersen's graph.

(2/Moderate) Modify Algorithm 3 to use the sets $U^{(i)}$ of coset representatives rather than the Schreier vectors. Note that the sets $U^{(i)}$ must be *extended* when $H^{(i)}$ is extended, for the same reason that the Schreier vectors had to be extended.

(3/Moderate) For the random Schreier-Sims method, how would you determine a "random" element?

## Bibliographical Remarks

The idea for the Schreier-Sims method is first presented in C. C. Sims, "*Computational methods in the study of permutation groups*", **Computational Problems in Abstract Algebra**, (Proceedings of a conference, Oxford, 1967), John Leech (editor), Pergamon, Oxford, 1970, 169-183. The paper indicates that Sims had implemented the method. The method is more fully presented in C. C. Sims, "*Computation with permutation groups*", (Proceedings of the Second Symposium on Symbolic and Algebraic Manipulation, Los Angeles, 1971), S. R. Petrick (editor), Association of Computing Machinery, New York, 1971, 23-28.

An early implementation is described in an unpublished manuscript : Karin Ferber, "*Ein Program zur Bestimmung der Ordnung grosser Permutationsgruppen*", Kiel, 1967, 8 pages. Ferber's implementation regarded the basic transversals $U^{(i)}$ as the generators $S^{(i)} - S^{(i+1)}$, and worked top-down rather than bottom-up. The transversals were used to limit the size of the strong generating set one usually gets when working top-down.

Another early implementation is described in J. S. Richardson, **GROUP : A Computer System for Group-Theoretic Calculations**, M. Sc. Thesis, University of Sydney, 1973. This thesis also suggests the extending Schreier-Sims method.

The Schreier-Todd-Coxeter-Sims method is due to Sims in an unpublished manuscript : C. C. Sims, "*Some algorithms based on coset enumeration*", Rutgers University, 1974. Sims had an experimental APL implementation of the method. In 1975, J. S. Leon produced a fullscale implementation of the algorithm and extensively investigated its performance. His work is described in J. S. Leon, "*On an algorithm for finding a base and strong generating set for a group given by generating permutations*", Mathematics of Computation **35**, 151 (1980) 941-974. Leon also develops the random Schreier-Sims method in this paper.

The author implemented the extending Schreier-Sims method in 1975 and the Schreier-Todd-Coxeter-Sims method in 1978. This work is described in G. Butler, **Computational Approaches to Certain Problems in the Theory of Finite Groups**, Ph. D. Thesis, University of Sydney, 1980, along with uses of the extending Schreier-Sims method. The extending Schreier-Sims method and some of its uses are also described in G. Butler and J. J. Cannon, "*Computing in permutation and matrix groups I : Normal closure, commutator subgroup, series*", Mathematics of Computation **39**, 160 (1982) 663-670.

An analysis of (essentially Ferber's implementation of) the Schreier-Sims method was first presented by M. Furst, J. Hopcroft, and E. Luks, "*Polynomial-time algorithms for permutation groups*", (Proceedings of the IEEE 21st Annual Symposium on the Foundations of Computer Science, October 13-15, 1980), 36-41, who also analyse some other group-theoretic algorithms.

Some references for the Todd-Coxeter algorithm are J. A. Todd and H. S. M. Coxeter, "*A practical method for enumerating cosets of a finite abstract group*", Proceedings of the Edinburgh Mathematical Society (2) **5** (1937) 26-34; J. J. Cannon, L. A. Dimino, G. Havas, and J. M. Watson, "*Implementation and analysis of the Todd-Coxeter algorithm*", Mathematics of Computation, **27** (1973) 463-490; and J. Neubüser, "*An elementary introduction to coset table methods in computational group theory*", **Groups-St Andrews 1981**, C. M. Campbell and E. F. Robertson (editors), London Mathematical Society Lecture Notes Series **71**, Cambridge University Press, Cambridge, 1982, 1-45.