EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

ISR-TH/68-37

COMPUTER PROGRAMMES FOR R.F. ACCELERATOR STUDIES

by

E. Keil and A.G. Ruggiero

Geneva - July 22, 1968

PS/6668
SUMMARY

In this report we present an up-to-date description of a set of programmes for studying the behaviour of particles in the R.F. system of a circular accelerator or storage ring.

We start with an explanation of the concepts upon which the stacking programmes are based. Then follows an outline of the criteria which have to be taken into account in the choice of parameters in order to achieve a good simulation of the physical phenomena to be studied. The remainder of this report is devoted to a formal description of the programmes, including their structure, the details of the procedures used, and the formats of input and output.
CONTENTS

INTRODUCTION
1. SIMULATION OF THE STACKING IN A STORAGE RING
   1.1. The stacking process
   1.2. The energy distribution in the stacked beam
   1.3. The stacking matrix
   1.4. Stacking at the "top" and stacking at the "bottom"
   1.5. The stacking efficiency $\eta$
   1.6. The distribution $V_B$ inside the bucket
   1.7. Conclusions

2. ANALYSIS OF THE CHOICE OF THE PARAMETERS ENTERING IN A STACKING PROCESS
   2.1. Parameters of the storage ring
   2.2. Other parameters and relations
   2.3. Completeness of the problem
   2.4. The scaled synchrotron space $(y - \psi)$
   2.5. The computer parameters
   2.6. Parameters for the stacking region
   2.7. Some limitations due to the change of the revolution frequency
   2.8. The time scaling
   2.9. The matching of the computer problem with the real problem
   2.10. Reducing the computer time

3. PROGRAMME DATI
   3.1. Arrangement of data
   3.2. Input
   3.3. Output
      3.3.1. Printed output
      3.3.2. Punched output
   3.4. Programme structure

PS/6668
4. PROGRAMME ASTACK
   4.1. General
   4.2. Notation
   4.3. Method and data preparation
      4.3.1. Specification of the accelerator
      4.3.2. Output specification
      4.3.3. Specification of initial conditions
         4.3.3.1. Simple input option
         4.3.3.2. Compound input option
         4.3.3.3. OTHER input facilities
   4.3.4. END card
   4.3.5. Conventions for input
   4.4. Description of output
   4.5. Essential information for the insertion of new subroutines
      4.5.1. Arrangement of data
      4.5.2. Programme structure
      4.5.3. The subroutines of the programme ASTACK
         4.5.3.1. Subroutine ANGLE, flow diagram
         4.5.3.2. Subroutine GAP, flow diagram
   4.6. Restrictions, files referenced, timing

5. PROGRAMME BSTACK
   5.1. Purpose
   5.2. Notation
   5.3. Method
   5.4. Input data
   5.5. Output
   5.6. Programme structure
   5.7. Restrictions
   5.8. Files referenced
   5.9. Timing
6. PROGRAMME CSTACK
   6.1. General
   6.2. Procedure
   6.3. Specification of data
   6.4. Programme structure
   6.5. Conventions for input data
   6.6. Output
   6.7. Restrictions
   6.8. Files referenced
   6.9. Timing

APPENDIX 1  - Subroutines for ANGLE
APPENDIX 2  - Subroutines for GAP
APPENDIX 3  - Example of compound data option
APPENDIX 4  - Subroutine OTHER

REFERENCES
Fig. 1a

Stacking Region

Fig. 1b
INTRODUCTION

The analytical theory of synchrotron phase space was worked out by Sessler and Symon\(^1\) and, in a different way, by Vogt-Nilsen\(^2\). It allows the derivation of a Hamiltonian, the calculation of trajectories in phase space, of bucket area, etc., even in slightly more complicated cases like R.F. voltages composed of several harmonics\(^4\) or the missing bucket case\(^3\).

However, the effect of an R.F. system on the distribution of stacked particles in synchrotron phase space has not yet been calculated analytically. Instead, this problem has been treated by numerical computation, first by Swenson\(^5\), who initiated the development of the stacking programmes ASTACK, BSTACK and CSTACK at CERN. These programmes or modifications of them were later on applied to other stacking computations by Keil and Nakach\(^6\) and by Ruggiero\(^7,8\).

In the following, Section 1 presents Swenson's method on which the three programmes are based.

Section 2 analyses the choice of the parameters that enter into the simulation of a stacking process and into the matching of the computer problem to the actual problem. Sometimes, the choice of these parameters can be tedious and time consuming, therefore, a fourth programme, DATI, described in Section 3, has been written to punch automatically the input deck for the ASTACK programme.

Sections 4, 5, 6 describe the stacking programmes ASTACK, BSTACK, CSTACK in that order. They were written by J.S. Hornsby in 1961\(^9\). These sections are essentially a compilation of his notes and include descriptions of changes made in the meantime.
The three programmes form a logical sequence in alphabetic order, programme BSTACK analyses ASTACK results, and CSTACK analyses BSTACK results. The communication between them is by magnetic tapes.

The appendix is a list of the subroutines available in programme ASTACK in order to describe the accelerator and the options for the input of the particle and gap initial conditions.

The four programmes are written in the FORTRAN dialect accepted by the SCOPE operating system on the CDC 6400 and CDC 6600 computers at CERN. They have been submitted to the CERN computer programme library.
1. SIMULATION OF THE STACKING IN A STORAGE RING

1.1. The stacking process

A schematic cross-section of the vacuum chamber of a storage ring is shown in Figure 1a.

At the right hand side a pulse is injected, trapped by the R.F. system and accelerated towards the stacking region which is already filled by a series of pulses.

There are two ways of stacking. The first is to move the beam from C to the top A of the stacked beam ("stacking at the top"). When the stacked beam is passed by the buckets, which contain the injected beam in synchrotron phase space, its energy is reduced, i.e., it is moved towards C for a distance which corresponds in energy to the total area of the buckets.

The second way ("stacking at the bottom") is to accelerate the injected beam from C to the bottom B of the stacked beam. Of course, the tail B will be moving towards C pulse after pulse.

The fundamental problem is, thus, to calculate the energy distribution of a stacked beam after N cycles for a given storage device.

One ideally expects that the energy spread of the stacked beam after N cycles is N times the area that a trapped single beam occupies in synchrotron phase space and that the energy distribution is rather rectangular. In practice, one finds a distribution which is sometimes very different from a rectangle, and a total energy spread larger than the ideal value.
For the quantitative description of these effects one defines the "stacking efficiency" \( n \). Our definition is the ratio of the number of the particles found experimentally in the ideal energy interval to the ideal number of particles which should be there. We assume that the beam is debunched between cycles and that there is no dependence of the energy distribution on the phase distribution. The stacking efficiency \( n \) can easily be evaluated when the distribution is known.

1.2. The energy distribution in the stacked beam

Denoting by \( f_N(E) \) the energy distribution after \( N \) cycles and by \( f_0(E) \) the distribution inside the moving buckets, we can write

\[
\frac{f_N}{N} + \frac{1}{N}(E) = \frac{f_N}{N}(E) + f_0(E)
\]

(1.1)

with \( f_N(E) \) a function of \( E \) and \( N \) which describes the change of the distribution \( f_N(E) \) due to the R.F. device during one cycle.

Swenson proposed to use for \( F_N(E) \) a vectorial representation. For this purpose he subdivided the stacking region into \( K \) channels as shown in Figure 1b.

Denoting by \( V_N \) the vector describing the distribution \( f_N(E) \) and by \( V_B \) the vector describing \( f_0(E) \), we write Eq. (1.1) in the following vectorial form

\[
V_N + 1 = AV_N + V_B
\]

(1.2)

where \( A \) is the "stacking matrix" which modifies the vector \( V_N \), the resulting vector corresponding to \( f_N'(E) \).

We must look for the elements of the matrix \( A \).
1.3. The stacking matrix

We take \( N \) particles with energy \( E \) and phase \( \vartheta \) uniformly distributed in the \( i \)-th channel of the stacking region. From the injection region around \( E = E_0 \) we move the buckets generated by the R.F. system linearly in time. The buckets enter the stacking region, pass the channel \( i \) which initially contains all the particles and reach the top of the first channel. During this operation the distribution of the particles will be modified.

We collect all the vectors \( V_j(i)(j=K,K-1,\ldots,2,1) \) representing the modified distribution of the particles at times in when the centre of the moving buckets is at the centre of the \( j \)-th channel. The \( s \)-th element of the vector \( V_j(i) \) (and of the vector \( V_N \)) is the number of particles contained in the \( s \)-th channel.

Thus we have \( K \) vectors. The information contained in them, each based on \( K \) energy channels, can be displayed in a square table having \( K \) rows (energy channels) and \( K \) columns (histograms).

We choose to shift the histograms in such a way that the elements of the principal diagonal are those contained in the \( i \)-th channel and we consider this array of coefficients as a \( K \times K \) matrix, \( (A) \). Further we make the reasonable assumption (see Ref. 5) that the vector \( V_j(i) \) does not depend on the absolute position of the channel \( i \) but only on the distance between the two channels \( i \) and \( j \). Thus the \( j \)-th column of the matrix \( A \) can be considered as the histogram obtained placing the \( N \) particles in the \( (k-j+1) \)-th channel and stopping the buckets at the \( (i+1) \)-th channel. We notice that the process of obtaining the final energy spectrum (or vector, \( V_N \)) resulting from any initial energy spectrum (or vector, \( V_N \)) is simply a matter of
matrix multiplication. Hence, the energy spectrum after the $(N+1)$-th cycle is given by Eq. (1.2).

A further suitable assumption (see Keil and Nakach\textsuperscript{5}) is that the vector $V_j(i)$ does not depend noticeably on the initial phase distribution. Then, taking the same initial value $\theta_0$ of the phase for all the particles, we can reduce the number of $N_p$ of the particles, reducing in the same way the computer time.

It has been verified (see Refs. 6, 7, and 8) by means of the computer that the largest modification of the energy distribution occurs when the buckets are rather close to the sample of particles, and that the distribution remains fairly unchanged when the buckets are either well below or well above the particles. This fact permits us to construct the so-called "expanded matrix" with an undefined number of columns which can be useful when, after several iterations, the length of the vector $V_N$ is larger than the horizontal size of the basic matrix $A$. To get the "expanded matrix" the first column is copied towards the left hand side and shifted up by one place, and the last column is copied towards the right hand side and shifted down by one place, and so on.

1.4. **Stacking at the "top" and stacking at the "bottom"**

Using the definition of stacking matrix $A$ given in the previous chapter, we see that the vector $V_{N+1}$ given by Eq. (1.2) represents the energy distribution after $(N+1)$ cycles of stacking at the "top".
When stacking at the "bottom" the distribution after $N$ cycles, denoted by $V_N$, is not passed by the buckets, then, before performing the product $A \cdot V_N$, the vector $V_N$ should be shifted by $Q$ channels for a total area in synchrotron phase space equivalent to the area of a bucket.

In the next section we shall call $\lambda$ the ratio of the area of a bucket to the area of a channel in synchrotron phase space. Thus when studying a stacking process at the "bottom", we must take $Q = \lambda$. *)

Because $Q$ must be an integer, this imposes a condition on the width of the channels.

The usual procedure is to keep 160 channels and to distribute uniformly the initial values of the energy of the particles in channel No. 80. For stacking at the "top", one is interested in the histograms obtained when the buckets passed the particles, thus one should retain the histograms from $j = 61$ to $j = 160$ included and set $i = 20$. For stacking at the "bottom" one should retain the vectors from $j = 1$ to $j = 100$ included when the particles were not passed by the buckets, and set $i = 80$.

*) As we will see in Section 6, where the programme CSTACK is described, the vector $V_N$ can be shifted more generally by $Q$ channels (up or down according to the sign of $Q$). When one wants to study a stacking process at the bottom, one must take $Q = \lambda$. But it may be possible that one requires a shift by a number of channels $Q$ less than $\lambda$ (see Refs. 7 and 8).
1.5. The stacking efficiency, $\eta$

Let us suppose that after $N$ iterations of the Eq. (1.2) we know the vector $V_N$, of which we denote the $s$-th element by $V_{N_s}$.

Following the definition of stacking efficiency $\eta_N$ after $N$ cycles given above we have:

for stacking at the "top"

\[ \eta_N = \frac{1}{N_a} \sum_{s = i}^{i + \lambda N - 1} V_{N_s}, \]  

(1.3)

and for stacking at the "bottom"

\[ \eta_N = \frac{1}{N_a} \sum_{s = i - \lambda N + 1}^{i} V_{N_s}, \]  

(1.4)

assuming that the vector $V_0$ is normalized to a (generally $a = \lambda$ or 1).

*) In the programme CSTACK, which performs the sums (1.3) and (1.4) $\lambda$ must be given as data and the following convention is used:

i) for stacking at the "top" $\lambda$ has the definition given in paragraph 1.4 and is always a positive quantity;

ii) for stacking at the "bottom" ($-\lambda$) must be read in.
For stacking at the top $Q$ is zero, and one can take also a non-integral value for $\lambda$; the last term of the summation (1.3) is then the appropriate fraction of the contents of the channel $i + \lambda N$.

1.6. The distribution $V_p$ inside the bucket

This can be computed assuming a uniform distribution of the particles in the area $A$ of the bucket. We cut the boundary of the bucket by parallels to the $\Theta$ axis, the distance in energy between two consecutive parallels being the width of a channel, and in such a way that the $\Theta$ axis is centred to the $i$-th channel. The contents of the channels is then proportional to the area limited by the boundary of the bucket and by the boundaries of the channels.

1.7. Conclusions

In agreement with the method exposed above the three computer programmes A-B-CSTACK adopt the following procedure.

$N_p$ particles are uniformly distributed in energy in the $i$-th channel, all with the same phase $\Theta_0$; this is part of the input data of the programme ASTACK. Each particle is followed as it circulates in the storage ring and crosses the R.F. cavities. The phase and the energy are printed out every time the centre of the moving buckets crosses the centre of a channel. The series of these times is another section of the input data of ASTACK.

The programme BSTACK collects the results of ASTACK and forms the histograms which will be communicated to CSTACK programme. This prepares the stacking matrix $A$ and makes the iterations (1.2) printing the energy spectrum and the stacking efficiency $n_N$ after $N$ cycles.
2. ANALYSIS OF THE CHOICE OF THE PARAMETERS ENTERING IN A STACKING PROCESS

2.1. Parameters of the Storage Ring

We will consider the case of stacking with moving buckets having constant \( I = \sin \theta \) and constant area \( A \) in the range of time and energy we are concerned with. A single accelerating cavity is present along the circular orbit.

The parameters which describe the storage ring are:

a) The R.F. voltage applied to the cavity

\[
V(t) = V \sin 2\pi f \int_0^t v dt
\]

(2.1)

where the amplitude \( V \) is constant in time and the frequency \( f \) is linearly modulated in time:

\[
v = v_0 + vt
\]

(2.2)

b) The equation relating the revolution frequency \( f(E) \) of a particle to its energy \( E \).

In order to have a constant area \( A \) of the bucket one should take (Ref. 11)

\[
f \frac{df}{dE} = \text{constant}
\]

(2.3)

from which follows, after integration,

\[
f(E) = f_0 \left[ \frac{2K_l E}{E_0} + (1 - 2K_l) \right]^{1/2}
\]

(2.4)
\[ K_1 = \frac{E}{f_0^2} \left( \frac{df}{dE} \right). \]  \hspace{1cm} (2.5)

It is negative above transition energy and positive below.

\[ f_0 \] is the revolution frequency at the energy \( E_0 \).

The following fundamental relation holds at any time

\[ v = \hbar f \] \hspace{1cm} (2.6)

where \( \hbar \) is an integer, the harmonic number.

From Eqs. (2.2), (2.4) and (2.6), we have

\[ t = \frac{\nu_0}{v} \left\{ \left[ 2K_1 \frac{E}{E_0} + (1 - 2K_1) \right]^{1/2} \right\} - 1 \]. \hspace{1cm} (2.7)

If we choose \( E_0 = 0 \), the formalism above breaks down.

In this case it is more appropriate to introduce the quantity

\[ H_1 = \frac{K_1}{E_0} = \frac{1}{f_0^2} \left( \frac{df}{dE} \right), \] \hspace{1cm} (2.5a)

from which and Eqs. (2.4), (2.7) we have

\[ f(E) = f_0 \left( 1 + 2H_1 E \right)^{1/2} \] \hspace{1cm} (2.4a)

and

\[ t = \frac{\nu_0}{v} \left\{ \left[ 1 + 2H_1 E \right]^{1/2} \right\} - 1 \]. \hspace{1cm} (2.7a)
2.2. Other parameters and relations

The stacking process is completely described by the following parameters:

a) the area $A$ of the moving bucket given by

$$2\pi A = 8\alpha(\Gamma) \frac{2V}{\sqrt{\pi h \left| \frac{df}{dE} \right|}}$$

(2.8)

where $\alpha(\Gamma)$ has been calculated by Sessler and Symon in Ref. 1.

b) The height $\delta E$ of the bucket

$$\delta E = \gamma_m(\Gamma) \frac{2V}{\sqrt{\pi h \left| \frac{df}{dE} \right|}}$$

(2.9)

with $\gamma_m(\Gamma)$ plotted in Ref. 1.

c) We call $N/2$ the number of revolutions which the synchronous particle executes in the interval of time in which it is accelerated from $E_0$ to $E_r$. We have

$$E_r - E_0 = \frac{NVT}{2}$$

(2.10)

d) We call $\Delta$ the ratio of the revolution frequency $f_0$ to the phase oscillation frequency $v_\phi$. The accelerating voltage $V$ is then given by

$$V = \frac{2\pi}{\cos \theta_s} \left[ \frac{1}{h(\frac{df}{dE})} \right] \frac{f_0^2}{\Delta^2}$$

(2.11)
e) Finally we have:

\[
\dot{\psi} = h \left( \frac{df}{dE} \right) V \Gamma .
\]  

(2.12)

2.3. Completeness of the problem

We note that the following two groups of parameters completely describe a stacking process:

i) \( V \) and ii) \( h \)

\[
\begin{align*}
\psi & \quad E_o \\
V_0 & \quad \Gamma, \alpha(\Gamma), \gamma_m(\Gamma) \\
\frac{df}{dE} & \quad \Delta \\
2\pi A & \quad N \\
E_p - E_o & \\
\delta & \\
\end{align*}
\]

When the values of the parameters of the second group are all given together with two parameters of the first group, the others can be computed using the Eqs. (2.8), (2.9), (2.10), (2.11), (2.12). Then also \( k_1 \) is completely determined.

In particular, we find the amplitude and frequency (2.1) of the accelerating R.F. voltage, the relation (2.4) between \( f \) and \( E \) and the time scaling (2.7).

We make the following observations:
a) The number $N/2$ of revolutions of the synchronous particle should be fixed a priori and should not be too large. In fact, the computer time is proportional to the number of steps (see ASTACK programme) which in the case of one cavity is exactly $N$.

b) When storage rings with a very small synchrotron frequency are studied, it is uneconomic to keep the large value of $\Delta$ because the computer time becomes too long. In this case one should use a smaller value of $\Delta$, e.g. $\Delta = 1000$ (see ref. 11).

2.4. The scaled synchrotron space $y - \psi$

According to Ref. 1 and Ref. 2, it is sometimes very useful to introduce the following scaled canonical variables

$$\psi = \theta + \frac{1 - \delta}{2} \pi, \text{ with } \delta = \left\{ \begin{array}{ll} +1 & \text{below transition energy} \\ -1 & \text{above transition energy} \end{array} \right. $$

and

$$y = \frac{1}{2\pi Q} \int_{0}^{E} \frac{dE}{z(E)} \frac{V}{Q} \quad (2.13)$$

with

$$2\pi Q = \sqrt{\frac{V}{2\pi h |\frac{d^2 E}{dE^2}|}} \quad (2.13a)$$

Introducing also a scaled time $\tau$ as independent variable

$$\tau = \frac{1}{\mathcal{K}} t \quad (2.13b)$$
with

\[ \delta = \sqrt{\frac{2\pi \hbar}{\int \frac{d\phi}{dE} \psi}} \quad (2.13c) \]

we have the constant Hamiltonian of a particle in the \( \psi - y \) plane

\[ H = I\psi - \cos \psi - \frac{1}{2} y^2 \quad (2.14) \]

associated with the canonical equations of motion

\[ \psi' = \frac{\partial \psi}{\partial \tau} = -y \quad (2.15a) \]

\[ y' = \frac{\partial y}{\partial \tau} = -(I + \sin \psi) \quad (2.15b) \]

2.5. The computer parameters

In Figure 2 we plot in the plane \((E, t)\) the energy \(E_s\) of the synchronous particle against \(t\).

Our goal is to investigate the energy distribution of a group of particles with mean energy \(E_s\) at \(t = 0\) when the moving buckets reach values of energy in the interval \((E_i, E_f)\).

We introduce the quantity

\[ \Omega = 1 - \frac{E_f - E_i}{E_f - E_i} \quad (2.16) \]
Fig. 2
which has the value 0.5 when $E_g$ is at the centre of the stacking interval $(E_i, E_p)$.

Other computer parameters are:

a) the number $N_h$ of histograms (or of channels) to be taken uniformly in the interval $(E_i, E_p)$.

b) We denote by $\Delta E$ the width of a channel, and by $\lambda$ the ratio of the area $A$ of a bucket in synchrotron phase space given by Eq. (2.8), to the area $A_c$ of a channel.

It is convenient to choose $\lambda$ not too large.

c) The number of $N_p$ of particles to be distributed uniformly at $t = 0$ in the interval $\Delta y_c$ of the channel around $E_g$, and between two or more separatrices in order to give the same weight to all the trajectories crossing that interval.

d) The number $\xi$ of intervals between two adjacent separatrices contained within that $y$ interval, $\Delta y_c$.

We take for all the particles the same value of the phase which corresponds to the phase $\vartheta_s$ of the synchronous particle.

The distance in the scaled energy variable $y$ between two consecutive separatrices around the mean value $y_g$ of the group of particles at $t = 0$, is, at the synchronous phase $\vartheta = \vartheta_s$,

$$\Delta y_s = \frac{2\pi \xi}{y_g} \quad (2.17)$$
and the height of a channel in y-coordinates is

\[ \Delta y_c = \frac{\delta a(\Gamma)}{\pi \lambda} \]  

(2.18)

then

\[ \zeta = \frac{y_c}{y_s} = \frac{8 a(\Gamma)}{\pi \lambda} \frac{y_s}{2 \pi \Gamma} \]  

(2.19)

It is convenient to take for \( \zeta \) an integer value.

Inserting Eq. (2.13) into (2.19) and assuming \( H_1 \ll 1 \), we get

\[ \zeta \lambda = \frac{E_k - E_s}{\nu \Delta a_\perp(\Gamma)} \]  

(2.20)

with

\[ a_\perp(\Gamma) = \frac{\pi \Gamma \sqrt{\cos \theta_s}}{8 a(\Gamma)} \]  

(2.21)

e) The ratio \( \sigma \) of the area \( S \) of the stacking region to the area \( A \) of a bucket is

\[ \sigma = \frac{S}{A} = \frac{N_h}{\lambda} \]  

(2.22)

This quantity should not be too small in order to put well in evidence the change of the energy distribution but also not very large in order to avoid a loss of details about the region in which the change mainly occurs. Generally one should take \( 10 \leq \sigma \leq 100 \).
From Eq. (2.8) and

\[ \Delta E = \frac{E_f - E_i}{N_h} \]

we have

\[ \frac{\alpha_{2}(\Gamma)}{\lambda} = \frac{E_f - E_i}{N_h} \]  \hspace{1cm} (2.23)

with

\[ a_2(\Gamma) = \frac{\hbar a(\Gamma)}{\pi^2} \sqrt{\left| \cos \Theta_s \right|} \]  \hspace{1cm} (2.24)

Thus, at last, we have three relations to be satisfied for the choice of the parameters in order to simulate the stacking process in a good approximation.

The three relations are the Eqs. (2.20), (2.22) and (2.23), in which the following fundamental parameters enter:

\[ \lambda, \zeta, \]
\[ E_g - E_o, \]
\[ \Delta, \]
\[ E_f - E_i, \]
\[ \sigma, \]
\[ N_h. \]
2.6. Parameters for the stacking region
These are \((E_f - E_i), (E_g - E_o)\) and \((E_f - E_o)\) related by

\[
\Omega(E_f - E_i) = (E_f - E_o) - (E_g - E_o) \quad (2.25)
\]

and by

\[
E_g - E_o = a_1(\Gamma) \nu A \lambda \zeta \quad (2.25a)
\]

\[
E_f - E_i = a_2(\Gamma) \Delta \frac{N \hbar}{\lambda} \quad (2.25b)
\]

\[
E_f - E_o = \frac{N \nu T}{2} \quad (2.25c)
\]

Inserting Eqs. (2.25a), (2.25b), (2.25c) into (2.25), we get the new parameter

\[
\varepsilon = \frac{N}{\Delta} = \frac{a_2}{\Gamma} \left(2\Omega \frac{N \hbar}{\lambda} + 2 \frac{a_1}{a_2} \lambda \zeta \right) \quad (2.26)
\]

which is close to the double of the number of phase oscillations in the computation.

A stacking process is mainly described by the parameter \(\Gamma\), (hence \(a_1\) and \(a_2\)). The quantities \(\lambda, \zeta, N, h\) are fixed at our convenience and according to the limitations mentioned above. Thus \(\sigma\) and \(\varepsilon\) are automatically known through Eqs. (2.22) and (2.26). Moreover, one of the two parameters \(N\) or \(\Lambda\) is also taken at our convenience, the other is then fixed by Eq. (2.26).
The problem is completely solved when also two of the quantities of the group i) in paragraph 2.3 are given. V being in this way computed, the energy scaling of the stacking region is obtained by Eqs. (2.25 a, b, c). From this it is easy to evaluate the boundaries of the energy channels and to give the initial conditions of the $N_p$ particles.

The initial value of the energy $E_o$ is completely at our choice.

2.7. Some limitations due to the change of the revolution frequency

The interval $E_f - E_o$ of the energies we are concerned with, given by Eq. (2.25c), should not be too large if the approximation

$$\sqrt{1 + 2\lambda(E_f - E_o)} - 1 + \lambda(E_f - E_o)$$

used to get Eq. (2.20) can still be applied. All the relations obtained above are verified, if the frequency $f(E)$ does not change very much in the interval $(E_f - E_o)$.

A criterion is to evaluate the change of the parameter $\lambda$ from the first channel to the last one. In fact, the value of this quantity is a function of the mean energy of the channel. Because the area $A_c$ of a channel in the synchrotron space $(W, \theta)$ is

$$A_c = 2\pi \Delta W$$

with

$$\Delta W = \frac{\Delta E}{2\pi f(E)}$$
we have

\[ \lambda = \frac{A}{A_c} = \frac{A}{\Delta E} f(E). \]

Because we impose that the change of \( \lambda \) in the range of energy \( E_f - E_i \) is small, we get again that the change of the frequency \( f \) should also be small, since

\[ \frac{\Delta \lambda}{\lambda} = \frac{\Delta f}{f} \quad (2.27) \]

2.8. The time scaling

The energy and the phase of the particles should be output at the time \( t_i \) (\( i = 1, 2, \ldots, \pi_i \)) at which the centre of the moving bucket crosses the centre of the \( i \)-th channel, numbered now from low energy.

The time \( t_i \) can be calculated from Eq. (2.7) or Eq. (2.7a) which show that there is no linear dependence between the time and the energy.

In particular from Eq. (2.7) we have:

\[ t_i = \frac{\nu_c}{v} \left\{ \left[ \frac{E_i}{E_0} \right]^{1/2} \left[ 2k_1 \frac{E_i}{E_0} + (1 - 2k_1) \right]^{-1} \right\}, \quad (2.28) \]

where \( E_i \) is the energy at the centre of the \( i \)-th channel.

For \( k_1 \) small we have the simpler expression

\[ t_i = \frac{\nu_c}{v} \frac{1}{k_1} (E_i - E_0) \quad (2.28a) \]
with $H_\perp$ given by Eq. (2.5a).

The time $\Delta t_i$ to cross the $i$-th channel is

$$\Delta t_i = \frac{v_0}{\sqrt{1 + 2H_\perp(E_i - E_o)}} \frac{H_\perp \Delta E}{1/2}$$  \hspace{1cm} (2.29)

and for $H_\perp$ small

$$\Delta t_i = \frac{v_0}{\sqrt{1 - H_\perp(E_i - E_o)}} \Delta E$$  \hspace{1cm} (2.29a)

$$= \frac{v_0}{\sqrt{1}} H_\perp \Delta E.$$  \hspace{1cm} (2.29b)

The cumulative error $\rho$ made in the time using the approximation (2.29b) is

$$\rho = \frac{v_0}{2\sqrt{1}} H_\perp^2 (E_f - E_o)^2.$$  \hspace{1cm} (2.30)

Thus the linear approximation can be used when $|\rho| \ll \Delta t_i$. Otherwise one should use Eq. (2.29a) or better, Eq. (2.29).

2.9. The matching of the computer problem with the real problem

The parameter introduced in the previous chapter of this section must be chosen also in such a way that the real stacking process is well simulated by the computer stacking process taking into account all the details of the experimental design.
Of course, the first parameter to match is $\Gamma$, and this must always be possible. With $\Gamma$ we get also $a(\Gamma), y_m(\Gamma), a_1(\Gamma)$ and $a_2(\Gamma)$.

The parameters $\lambda, \zeta, N_h$ (and the number of the particles $N_p$) can be chosen at our convenience for every case, their choice does not influence directly the match of the two problems.

Swenson suggested also the matching of the $\Delta$'s in order to reproduce the discontinuity of the change in energy when the particle crosses a cavity. This matching is generally uneconomical because it requires a long computer time which can be estimated by the number $N$ of steps using Eq. (2.26).

If $N$ is too large, one has two possibilities:

a) retain the matching of $\Delta$ but choose the initial conditions of the particles such as to reduce the computer time; this will be done in the next chapter;

b) abandon the matching of $\Delta$, but then we must look for another way to get the necessary conditions for a suitable matching of the two problems.

It is clear that when the matching of $\Delta$ is possible, it is convenient to match also all the other physical parameters.

In this paragraph we will fulfil the requirement of the second possibility b).
We assume that the two problems are equivalent when the trajectory of a particle in the synchrotron space \((W,\phi)\) is the same in the two cases.

From Ref. 3) the Hamiltonian of a particle with any amplitude modulation \(Vg(t)\) of the R.F. voltage is

\[
\mathcal{H}^* = \frac{\Theta w}{\hbar \Omega'_s} - \frac{\hbar \Omega'_s}{2} W^* W^2 + \frac{a V}{2\pi} \int g(\phi) \sin \phi \, d\phi .
\]

The parameters are:

\[
\omega = 2\pi v,
\]

\[
\hbar \Omega'_s = 4\pi^2 \hbar f \frac{df}{d\phi}, \quad \text{constant}
\]

\(\hbar = \text{it may be a parameter of the modulation } g(\phi)\)

\(V\)

\(\theta\)

\(W^* = W - W_s, \quad W_s \text{ belongs to the synchronous particle}\)

We see that the frequency \(f\) does not enter alone but in the form \(E/f\) through \(W^*\). Then, if we scale the frequency and the energy by the same amount and we retain all the other variables, we see that the Hamiltonian and the trajectory are the same in the two cases.

\(*\) We see that in the normal case \(g(\phi) = 1\), since \(\hbar\) enters always in the form \(\hbar \Omega'_s\), the matching of \(\mathcal{H}\) is not necessary but rather the matching of \(\hbar \Omega'_s\). It is clear that it can be very useful to take \(\hbar = 1\) in this case.
Thus, when the real $\Delta$ is too large, a convenient procedure is to keep the computer $\Delta$ about equal to 1000, in this way $\Delta$ is scaled by an amount, say, $\alpha$. The energy and the frequency $f(E)$ will be scaled by the same amount $\alpha$. We take one of the quantities

$$V$$
$$f \frac{df}{dE}$$

$A$

$\delta E$

$E/E$ = relative change in time of the energy
equal to that of the actual case and through Eqs. (2.8), (2.9), (2.10), (2.11) and (2.12) all the others can be easily computed.
The new values that will be introduced in the subsequent computations are equal to the actual values.

2.10. Reducing the computer time

Also in the case in which the matching of $\Delta$ is abandoned and $\Delta \sim 1000$ is taken the computer time may be too long.

This is due to the fact that the time the computer requires to simulate a revolution of a particle in the accelerating ring is much longer than $1/f_0$.

To reduce the computer time one can, for example, try to determine the distribution in energy and in phase of the particles at an advanced time $t$ when the distribution at $t = 0$ is known. This is only possible by solving the equations of motion exactly.
However, it can be done approximately.

As mentioned in paragraph 1.3 and as shown in Refs. 6), 7) and 8), when the computer parameters are chosen satisfying the necessary conditions, we can assume that the energy distribution of the particles is nearly unchanged at the time $t_i$ corresponding to the energy $E_i$ in Figure 2 and, surely, this is better verified at a time $t < t_i$.

We assume thus, that at any time $t < t_i$ the energy of a particle is practically the same as at $t = 0$. Of course, the phase distribution will be modified, but it is not difficult to evaluate it at the time $t$ knowing that at $t = 0$ all the particles had the same phase $\Theta = \Theta_0$, the synchronous value.

Because the buckets are moving linearly in time, we have approximately for a particle

$$ y = y_o - y'\tau $$

(2.31)

$\tau = 0$ corresponding to $t = 0$.

Inserting Eq. (2.31) in the Eq. (2.15a) we have

$$ \psi' = -y $$

$$ = y'\tau - y_o $$

then

$$ \psi = \frac{1}{2} y'\tau^2 - y_o \tau + \psi_s $$

(2.32)
But from Eq. (2.13), if the change of $f$ is small, it is

$$
y_0 = \frac{E - E_0}{2\pi f_0 Q},
$$

and

$$
y' = \frac{VTK}{2\pi Q}.
$$

Inserting the expressions of $Q$ and $K$, Eqs. (2.13a and c) and using Eq. (2.12), we have

$$
\phi = \frac{1}{\hbar} t^2 - \frac{E - E_0}{f_0} \left( f \frac{df}{dE} \right) t + \phi_s,
$$

(2.33)

where $\phi = \theta/(2\pi \hbar)$ is the phase variable as used in the ASTACK programme. Eq. (2.33) gives the phase $\phi$ of a particle with initial value $E$ and $\phi_s$ at the time $t$.

Note, that Eq. (2.31) is exact apart from an oscillatory term with amplitude inversely proportional to $y$, thus, if the bucket is well below the particles it can be neglected in Eq. (2.31), but it can give a substantial contribution in Eq. (2.33), where the phase $\phi$ is continuously normalized to unity. But neglecting the oscillating term in Eq. (2.31) is simply equivalent in a small shift of the initial phase of the particle. This shift depends on the energy, thus, Eq. (2.33) can be assumed to give the exact transformation of the phase, when the oscillating term is taken into account, of particles distributed in the proximity of $\phi = \phi_s$ with a small spread.

In the ASTACK programme a routine is available (see Appendix h) which calculates according to Eq. (2.33) the new phase distribution at a fixed time $t$. 

PS/6668
3. **PROGRAMME DATI**

The user of the ASTACK programme should prepare the input data following the suggestions and the requirements presented in the two previous sections. This is, in principle, always possible and really not conceptually difficult. Nevertheless, the high number of parameters and of the restrictions and relations bounding them make this part of the work tedious and time consuming, especially in the case in which several groups of parameters with different values of \( \lambda, \zeta, N_a \) should be compared.

The programme DATI which is described in the present section has been written just to facilitate the search of the best group of parameters. One of its purposes is to punch, when requested, the cards which will form the ASTACK input deck. For this purpose, it requires a very small number of data.

New subroutines are easily inserted into the programme DATI for the following cases:

i) punch the data cards for the BSTACK programme,
ii) evaluate the energy distribution in a bucket with given \( \Gamma \) and uniformly filled by particles,
iii) punch the data cards for the CSTACK programme.

The automatic punching of the cards is very useful when, as is usually the case, the input deck is very long.
3.1. **Arrangement of data**

The programme is based on a fundamental input vector A(I) of 8 variables and on a fundamental output vector B(I) of 7 variables.

The variables of the vector A(I) are

\[ A(1) = \Gamma \]
\[ A(2) = \Delta \]
\[ A(3) = \alpha \]
\[ A(4) = \lambda \]
\[ A(5) = N \]
\[ A(6) = E_0 \]
\[ A(7) = \alpha(\Gamma) \]
\[ A(8) = y_m(\Gamma) \]

The variables of the vector B(I) are:

\[ B(1) = V \]
\[ B(2) = \bar{\zeta} \]
\[ B(3) = f_0 \]
\[ B(4) = f \frac{df}{dE} \]
\[ B(5) = 2\pi A \]
\[ B(6) = E/E_0, \text{ relative change in time of the energy of the synchronous particle} \]
\[ B(7) = \delta E/E_0, \text{ height of the bucket.} \]

Further input variables are:
NPAR = \( N_D \), number of particles
NHIST = \( N_H \), number of outputs or histograms
NEPA = \( \xi \)
NREF = \( N_r \), number of the channel containing the particles at \( t = 0 \).

Further output variables are:

\[ BB(1) = a_1(\Gamma) \]
\[ BB(2) = a_2(\Gamma) \]
\[ SIGMA = \sigma = N_H/\lambda \]
\[ EPSILON = \varepsilon = N/\Delta \]
\[ EINIT = E_i \]
\[ ENER = E \]
\[ ENEF = E_f \]
\[ EINT = E_f - E_i \]
\[ DE = \text{height of a channel} = \Delta E \]
\[ DTC = \text{time to cross a channel} \]
\[ COMFAC = \frac{1}{r^2} \left( \frac{df}{dE} \right) \]

The components of the vector \( T(i) \) are the times \( t_i \) at which output of the results is requested.

3.2. Input

The input deck is formed by 5 cards which contain in that order:

i) The vector \( A(I) \) in the sequence mentioned above. The format is FORMAT (8 E 10.4)

Either \( A(1) = \Delta \) or \( A(5) = N \) should be blank.
ii) The necessary data to prepare and punch, when requested, the input deck of the ASTACK programme. When punching is not requested, this card must contain all blanks, otherwise it has 8 variables according to the format

\[
\text{FORMAT (} 5 \text{ I 10, 2 f 10.4, I 2) .}
\]

The 8 variables are in the order:

- **NS**: SENARO number (see ASTACK programme)
- **NB**: number of the record after which the results of ASTACK will be recorded onto the output tape (see ASTACK programme),
- **JOB**: has the following values:
  - = 0, the normal procedure to compute the energy and the phase of a particle starting from the time \( t = 0 \) is requested
  - > 0, the particular procedure to reduce the computer time should be followed
- **L1, L2**: first and last record of the output tape to be printed (see ASTACK programme)
- **AML, AM2**: generally they are zero. They are used for the missing buckets procedure. \( AML = A_1 \cdot A(3) \)
  \[ AM2 = A_2 \cdot A(3), \text{ (see appendix 2)} \]
- **IND**: it is zero when \( JOB = 0 \), otherwise it is a positive integer or zero with the effect to compute the time

\[
t_o = t_i - \text{IND} \cdot (t_f - t_i) \text{ (see Fig. 2)}
\]
from which the energy and the phase of a particle are computed following the particular procedure to reduce the computer time (see Appendix 4).

iii) In this card the absolute value *) of one of the variables \( B(I) \) is given preceded by the number \( I \) according to the format

\[ \text{FORMAT (I6, I4, E13.7)} \]

iv) The same as before but giving the absolute value *) of another parameter \( B(J) \) preceded by the number \( J \).

v) The last card contains in the order NPART, NHIST, NETA, NREF, TRANS with the format

\[ \text{FORMAT (I4 I10, F4.1)} \]

\[ \text{TRANS = -1 above the transition energy} \]
\[ \text{1 below the transition energy.} \]

*) We recall that \( \dot{\gamma} \) and \( f (df/dE) \) should be specified with their sign, but this will be given by the word TRANS in the following.
There are only two restrictions on the cards no 3 and 4. The two parameters $B(I)$ (in the third card) and $B(J)$ (in the card no. 4) must always be taken in such a way that $I < J$.

Further the following combinations are incompatible

\[
\begin{align*}
I &= 1 \quad \text{and} \quad J = 7 \\
I &= 2 \quad \text{and} \quad J = 3.
\end{align*}
\]

3.3. Output

We have the printed output and the punched output.

3.3.1. Printed output

On the first page the following quantities are given line by line:

- $A(1)$
- $A(2)$
- $A(3)$
- $A(4)$
- $A(5)$
- $A(6)$
- $A(7)$
- $A(8)$

\[ N_p, N_{d}, N_r \]

\[ a_1(r), a_2(r) \]

These two quantities are given by the vector $BB(I)$

\[ \xi \]

\[ \sigma \]

\[ \epsilon \]

Energy scale $E_i$, $E_g$, $E_r$ and $(E_r - E_i)$

PS/6668
B(1)
B(2) (with sign)
B(3)
B(4) (with sign)
B(5)
B(6)
B(7)

Height of a channel = (ΔE)
Time to cross a channel = (Δt)
Maximum error = (ε)
Compaction factor = (H_)
Stable phase phi = (arc sin Γ = φ_s)
Stable phase teta/2π = (φ_s/2πh)
Frequency variation = (Δf/f)

On a new page introduced by the label "CHANNEL", "LOWER BOUND", "UPPER BOUND", the lower and the upper energy boundaries of the channels are given.

It follows the list of the number of the output (or histogram), the corresponding synchronous value of the energy and the time at which the output should occur.

The programme also calculates the parameters for a stacking process with a certain number of suppressed buckets (78). In this case the last data printed in a single line

\[ ΦT = (φ_s/2π h) \]

gives the first of the h values of the phase φ_s/2π h in one interval of φ_s which corresponds to an existing bucket.
The \( N_p \) particles will be distributed uniformly in energy at this value of \( \phi_s/2\pi \ h \) (TET).

For a normal stacking process TET is just the first value of \( \phi_s/2\pi \ h \) which is already given and has so significance in the subsequent calculation.

3.3.2. Punched output

When requested (JOB > 0 in the input data) the programme furnishes the input deck for the ASTACK programme for the case of a single cavity with \( V \) and \( \psi \) constant and uses the law of the change of the revolution frequency \( f \) against energy given by Eq. (2.4).

Besides it uses one of the two subroutines \( NL = 5 \) and \( NL = 6 \) (see Appendix 2) for the step GAP.

3.4. Programme Structure

The programme starts reading all the input deck. Then it computes \( a_{1}(\Gamma), a_{2}(\Gamma), \sigma \) and \( \varepsilon \). By \( \varepsilon \) and using Eq. (2.26) it computes also that of the two parameters \( \Delta \) and \( N \) left blank in the first data card. At this point the programme prints the vector \( \Delta(\Gamma), N_p, N_u, N_r, a_{1}(\Gamma), a_{2}(\Gamma) \) and \( n, \sigma, \varepsilon \).

The second part of the programme consists of computing all the elements of the vector \( B(\Gamma) \) starting by the ones given in the data cards and using the Eqs. (2.8), (2.9), (2.11), (2.12) and

\[
\frac{E}{E_0} = \frac{V\Gamma}{E_0}.
\]
For this purpose the programme calls the I-th of the 6 sub-routines V1, V2, V3, V4, V5, V6 where I is given in the third data card.

The energy scaling is computed from Eqs. (2.25a, b, c) and it is then output followed by the vector B(I). Note that the two parameters \( f (\frac{df}{dE}) \) and \( \phi \) are computed with their sign.

At the end of the second part of the programme \( \Delta E, \Delta t, \phi, \%
\)

\( H, H_a \) are calculated and printed. At last it calculates the change of the frequency \( f \) in the energy interval \( E_f - E_0 \) which can be useful to judge the accuracy of the simulation of the stacking process as mentioned in the previous section.

In the third part the boundaries of the energy channels and the output times are computed. The punching of the cards forming the input deck of the ASTACK programme is executed by the subroutine SCHEDE.

At this point the programme is ready to read new data cards.

4. PROGRAMME ASTACK

4.1. General

This programme is based on ideas incorporated in a programme prepared at MURA (TTT, programme 39) for R.F. studies. It traces the energy and relative phase of a particle, injected with given initial energy and phase, as it circulates in an accelerator, taking account of the effect of R.F. gaps which the particle meets during each orbit.

PS/6668
Each revolution of a particle is simulated by a given series of steps. A step is one of the following fundamental operations:

i) the particle traverses a circular section of the accelerator. Its energy is constant. The current time increases following a given law which is generally a function of the energy and may be different whether or not betatron oscillations are taken into consideration (ANGLE).

ii) The particle crosses a gap. Its energy and phase are changed following a given law (GAP).

iii) The particle has a lumped loss either of its energy or of its transverse momentum. A transverse displacement can also occur (FOIL).

Output from the programme consists of a table of values of energy, phase and time, occurring after specified numbers of steps or after specified moments of time. The input data and the results are written onto magnetic tape in such a way that the BSTACK programme (or other programmes) may easily process and digest the results of simulating the behaviour of numbers of particles.

The programme has been designed with the intention that the functions which represent the behaviour of the particle as it traverses an angle or crosses a gap can easily be changed, in accordance with the physical process taking place, by adding coding to the appropriate subroutines.
Betatron oscillations, and energy loss in foils may also be studied in this way. However, the necessary routines are not included in the present version of the programme.

4.2. Notation

\begin{align*}
t & = \text{time (in seconds or any other convenient unit)} \\
E(t) & = \text{particle energy (any convenient unit)} \\
x(t) & = \text{particle coordinate of betatron oscillations} \\
p(t) & = \text{particle momentum of betatron oscillations} \\
\phi_{j,N} & = \text{phase of particle relative to the } j\text{-th oscillator on the } N\text{-th revolution (in units of } 2\pi \text{ radians)} \\
t_{j,N} & = \text{time of arrival of the particle at the } j\text{-th oscillator on the } N\text{-th revolution} \\
\Delta t_{j,N} & = \text{time between } N\text{-th and } (N-1)\text{-th arrival at the } j\text{-th oscillator} \\
V_j(t) & = \text{peak energy gain per traversal of the } j\text{-th oscillator (in the same units as } E(t)) \\
v_j(t) & = \text{frequency of the } j\text{-th oscillator (cycles per unit of time)} \\
\{E_0, c_0, x_0, p_0, \phi_{j,0}, t_{j,0}\} & = \text{initial values for the particle} \\
\{h_j\} & = \text{harmonic number of the } j\text{-th oscillator, the ratio of } v_j(t) \text{ to the revolution frequency of the synchronous particle.}
\end{align*}
We define the phases, \( \phi_j \), such that they increase by about one in one revolution. Hence, the harmonic numbers \( h_j \) appear explicitly in the expressions (4.1) and (4.2) for the phase and energy change.

4.3. Method and data preparation

Data can be divided into three classes:

i) specifications of the accelerator

ii) output specifications,

iii) specifications of particle and gap initial conditions.

All data should be given in a seven column table with the following layout on cards and the following formats:

<table>
<thead>
<tr>
<th>M</th>
<th>IDENT</th>
<th>N1</th>
<th>N2</th>
<th>VALUE</th>
<th>N3</th>
<th>N4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>card columns</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>6 - 11</td>
<td>15 - 20</td>
<td>25 - 30</td>
<td>36 - 50</td>
<td>55 - 60</td>
<td>65 - 70</td>
</tr>
<tr>
<td></td>
<td>FORMAT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>II</td>
<td>A6</td>
<td>I6</td>
<td>I6</td>
<td>E 15.8</td>
<td>I6</td>
</tr>
</tbody>
</table>

The M column is used to distinguish simple and compound data cards as described below. IDENT contains an identifying name describing the type of card or blanks. The N1, N2, N3 and N4 columns contain integers, and the VALUE column a floating point number.

PS/6668
4.3.1. Specification of the accelerator

The specification of the accelerator is given in a sequence of cards which starts with SENARO and ends with CURTIN. Between them may be cards with ANGLE, GAP, FOIL, and/or blanks in the identification column. All the cards without identification name must have blanks in column N3.

SENARO card

It must be the first card in the deck and indicates that cards describing the accelerator follow. An identifying number should be written in the NL column for later use by the BSTACK programme.

A number in the N4 column may be used to specify the position of the results on the output tape. If N4 = 131071 results will start at the beginning of the tape. If N4 is any other number it is interpreted as the last "record" number to be retained and writing will start after that "record". However, if an END "record" is found before the specified "record" number is reached writing will start at the END "record" which will be overwritten. If column N4 is left blank writing will start overwriting the first END encountered. It is thus possible to restart writing the output tape at almost any point without losing correct information on the tape. The printed output to be described below is specifically designed to facilitate this operation. The precise meaning of the term "record" is also given there.

In the case in which there are several cavities and the output value of the phase is requested with respect to the j-th cavity, the number j must be written in column N3. If this column is blank, the value of the phase with respect to the last cavity encountered is printed.
ANGLE card

An ANGLE represents a fraction of the circumference of the accelerator given in units of $2\pi$ radians in the VALUE column. If betatron oscillations are not studied, as is usually the case, the current time is increased by the interval it takes the particle to traverse that angle. The formula for evaluating this time interval is specified by N1; a list of the available values of N1 and of the corresponding formulae is given in Appendix 1. N2 specifies the formula used for transforming betatron oscillations. In the present version of the programme it is not used, and should be set $N2 = 1$.

After the ANGLE card follow some cards with numbers in the VALUE column and blank elsewhere. They specify the parameters for the evaluation of the time interval as specified in Appendix 1.

GAP card

A GAP represents an R.F. accelerating cavity which changes the energy of the particle. The following computations are performed. The phase of the particle relative to the j-th gap is calculated from

$$\phi_{j,N} = \phi \quad \text{for} \quad \phi \geq 0$$

$$\phi_{j,N} = 1 + \phi \quad \text{for} \quad \phi < 0$$

where

$\phi = \text{fractional part of}$
\[
\left\{ \phi_{j,N} - v_j(t_{j,N} - \frac{1}{2} \Delta t_{j,N}) \Delta t_{j,N} / \hbar_j \right\} \quad (\ast)
\]

(4.1)

And the energy is changed into:

\[
E = E + v_j(t_{j,N}) \sin (2\pi \hbar_j \phi_{j,N}).
\]

(4.2)

The formula for computing the voltage \(v_j\) is specified by NL, that for computing the frequency \(v_j\) by N2. The values of NL and N2 permitted by the programme and the corresponding formulae for \(V(t)\) and \(v(t)\) are given in Appendix 2. The VALUE column must be blank.

After the GAP card follow some cards with numbers in the value column and blanks elsewhere which specify the parameters for computing \(V(t)\) and \(v(t)\) as given in Appendix 2. Note that formula (4.1) can be used only in the case in which \(v_j\) changes very little during the time \(\Delta t_{j,N}\).

**FOIL card**

FOIL cards are recognized by the programme but cause an error exit since the FOIL subroutine is a dummy in the present version of the programme.

**CURTIN card**

This card terminates the description of the accelerator.

There may be up to 500 cards from the SENARO to the CURTIN card.

\[(\ast) \quad v_j(t_{j,N} - (1/2) \Delta t_{j,N}) \text{ means the frequency } v_j \text{ evaluated at the time } t_{j,N} - (1/2) \Delta t_{j,N} \]
4.3.2. **Output specification**

Output may be requested in one of three ways. A step as used below is defined to be the process of carrying out the transformation associated with one ANGLE, one GAP or one FOIL. Thus, going once round the accelerator consists of a number of steps.

i) Request output to occur every \( m_j \) steps during the first \( M_j \) steps, then every \( m_2 \) steps during the next \( M_2 \) steps ... etc., where \( m_j = 0 \) is taken to mean no output for \( M_j \) steps. This output is obtained by

**COUNT cards**

with \( j \) in the \( N_2 \) column, \( N_j \) in the \( N_3 \) column, and \( m_j \) in the \( N_4 \) column. There may be up to 500 COUNT cards in the deck.

ii) Request output to occur each time \( t \) becomes \( > T_i \), where the list of steadily increasing values of \( T_i \) is given on

**TIMOPT cards**

with \( i \) in the \( N_2 \) column and \( T_i \) in the VALUE column. The \( N_1 \) column must be blank. There may be up to 500 TIMOPT cards in the deck.

iii) Output after time intervals \( T_i \) with

\[
T_i = T_F (\Delta T) T_L
\]

is obtained by using a sequence of three

**TIMOPT cards**

with \( N_1 = 1 \) \( N_2 = 1 \) \( \text{VALUE} = T_F \)

\( N_1 = 1 \) \( N_2 = 2 \) \( \text{VALUE} = \Delta T \)

\( N_1 = 1 \) \( N_2 = 3 \) \( \text{VALUE} = T_L \).
4.3.3. Specification of particle and gap initial conditions

For the input of the particle and gap initial conditions two options exist: a simple one for input of initial conditions for one particle at a time, and a compound one for input of systematically varied initial conditions for several particles. The data cards for the two options are distinguished by the content of the M column: blank cards belong to the simple option, cards with a "1" belong to the compound option.

4.3.3.1. Simple input option

JOBNUM card

is used to label a particle for later identification by the BSTACK programme. Write an identifying number in the NL column.

ENERGY cards
TIMZRO cards
XZERO cards (*)
PZERO cards (*)

contain the initial values of $E_0$, $t_0$, $x_0$ or $p_0$, respectively, in the VALUE column, the N2 column must be blank.

FIGAP cards
TIMGAP cards

contain the initial conditions $\phi_{j,0}$ or $t_{j,0}$ of the j-th gap in the VALUE column, and the gap number j in the N2 column.

The above mentioned series of cards must be followed by

*) These cards will be used only if the betatron oscillations are taken into account.

PS/6668
BEGIN card

It terminates the input of data for a particle and starts the computation. On completion the computer will be ready to read further data.

Only those parameters which change between particles need be specified.

4.3.3.2. Compound input option

Supposing we want to study a number of particles with different starting conditions in an accelerator with $N_g$ gaps. Supposing first that all the particles have the same initial energy $E$. The initial phases of the first particle relative to the gaps we denote by

$$(\phi_1)_1, (\phi_2)_1, \ldots, (\phi_{N_g})_1$$

This set can be considered as a vector $\phi_1$ with $N_g$ components. Similarly we have for the $j$-th particle

$$(\phi_1)_j, (\phi_2)_j, \ldots, (\phi_{N_g})_j$$

represented by a vector $\phi_j$.

Let the number of such vectors be $N_p$. Then the set of all starting conditions is represented by $(E, \phi_j), j = 1, 2, \ldots, N_p$.

If now there is a set of initial particle energies $E_i, i = 1, 2, \ldots, N_E$, then the set of all starting conditions is $(E_i, \phi_j), i = 1 \ldots N_E, j = 1 \ldots N_p$, i.e. $N_E N_p$ particles are studied altogether.

The set of $\phi_j$ may be pictured as a matrix as follows:
The compound input option permits the specification of each row of the matrix, i.e. the starting conditions for each gap, and the specification of each particle energy. The programme then takes as initial conditions all combinations of $E_i$ with $\phi_j$ in the order

$$(E_1, \phi_1), (E_2, \phi_1), \cdots (E_N, \phi_1), (E_1, \phi_2), (E_2, \phi_2), \cdots (E_N, \phi_2)$$

$$(E_1, \phi_3), \cdots E_1, \phi_N \cdots (E_N, \phi_N).$$

The table below shows how the sets of $E_i$ or $(\phi_j), K = 1 \cdots N_F$ may be specified by a sequence of cards.

There are two alternative ways of specifying the set of energies $E_i$ (sequences 1 and 2), and of specifying a row of the matrix of phases (sequences 3 and 4). They are distinguished by the number in the NL column. NL = 1 means that the values are regularly spaced, i.e.
for sequence 1: \( E_i = E_1 + (i - 1) \Delta E \quad i = 1 \ldots N_E \)

for sequence 3: \( (\phi_{K})_j = (\phi_1)_j + (K - 1) \Delta \phi_j \quad K = 1 \ldots N_F \)

\( N_1 = 2 \) means that the whole set of energies \( E_i \) (sequence 2) or phases for the \( j \)-th gap (sequence 4) are specified in the value column of this and subsequent cards.

The number of the gap \( j \) is given in the \( N_2 \) column of the FIGAP cards.

<table>
<thead>
<tr>
<th>M</th>
<th>IDENT</th>
<th>N1</th>
<th>N2</th>
<th>VALUE</th>
<th>N3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ENERGY</td>
<td>1</td>
<td></td>
<td>( E_1 )</td>
<td>( N_E )</td>
</tr>
<tr>
<td>1</td>
<td>ENERGY</td>
<td>2</td>
<td></td>
<td>( E_1 )</td>
<td>( N_E )</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>( \Delta E )</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>( E_2 )</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>( \ldots )</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>( E_{N_E} )</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>FIGAP</td>
<td>1</td>
<td>( j )</td>
<td>(( \phi_1 ))_j</td>
<td>( N_F )</td>
</tr>
<tr>
<td>1</td>
<td>FIGAP</td>
<td>2</td>
<td>( j )</td>
<td>(( \phi_1 ))_j</td>
<td>( N_F )</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>( \Delta \phi_j )</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(( \phi_2 ))_j</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>( \ldots )</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(( \phi_{N_F} ))_j</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>BEGIN</td>
<td>( N_J )</td>
<td></td>
<td></td>
<td>( N_J )</td>
</tr>
</tbody>
</table>
BEGIN card

It must also have a "1" in the M column and the number of gaps N_o in the N2 column. The value N_j in the N3 column is the job number (JOBNUM) given to the first set of initial conditions. This number is increased by one for each subsequent set of initial conditions.

An example of the use of the compound option is given in Appendix 3.

Within each group of ENERGY, FIGAP, BEGIN cards all cards must be either of the simple, or of the compound type. Thus, if there is only one energy value but a set of phases specified by compound data cards, also the energy must be input by compounded data cards, and conversely for one phase and a set of energies.

4.3.3.3. OTHER input facilities

In addition to the acceptable data described above the following sequence of cards is also accepted.

<table>
<thead>
<tr>
<th>row</th>
<th>M</th>
<th>IDENT</th>
<th>N1</th>
<th>N2</th>
<th>VALUE</th>
<th>N3</th>
<th>N4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>OTHER</td>
<td>3</td>
<td>ID</td>
<td>E_0</td>
<td>N_T</td>
<td>N_F</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>V_0</td>
<td>n_E</td>
<td>m_E</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>at</td>
<td>at</td>
<td>n_F</td>
<td>m_F</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>n_E</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>n_F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ψ</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

PS/6668
This sequence defines a set of initial conditions as follows:

\[ (\phi_o)_i = \frac{2i - 1}{2N_p}, \quad i = n_p(1) m_p \]

and

\[ (E_o)_n,i = E_o - (at)(\phi_o)_i + V_o \cos \left[ \frac{\pi (2n - 1)}{2N_T} \right], \quad n = n_E(1) m_E. \]

However, the initial condition so defined is only used provided it lies within a sinusoidal envelope in \( E = \phi \) space defined by

\[ E - \bar{E} \sin \left[ \pi(\phi_o,i + \psi) \right] \leq (E_o)_n,i \leq E + \bar{E} \sin \left[ \pi(\phi_o,i + \psi) \right] \]

The job number associated with each particle is

\[ \text{job number} = \text{ID} + 100(i - 1) + n. \]

To deal with initial conditions which are generated according to some special law, the identifying word OTHER can be used. The special law required is defined by the value in the NL column. At present only NL = 3 and NL = 4 are meaningful. NL = 3 as shown above and NL = 4 as shown in Section 2 and Appendix 4. The option NL = 4 can be used when the computer time is too long and must be reduced according to the particular assumptions of Section 2.

Do not use the value 1 or 2 for NL.

Do not terminate the OTHER card series with a BEGIN card.
4.3.4. **END card**

It terminates the computations, empties the output buffer and writes an end-of-file on the output tape.

If VALUE ≠ 0.0, a number of lines from the record N1 to the record N2 inclusive, is copied from the output tape onto the printed output.

4.3.5. **Conventions for input**

If any of the initial values \(E_0, t_0, x_0, P_0, \phi_0, t_0\) are not specified at the beginning of a run they are taken to be zero. Between successive SENARO cards all initial values are carried forward from one calculation to the next but updated by any new values of \(E_0, t_0, \ldots \) which may be read in. However, when a new SENARO card is read, all initial values are reset to zero, non-zero values must be respecified.

The output options specified for one job are carried forward to the next job unless new output options are specified, or unless a new SENARO card is read, in which case new output options must be specified.

Only cards with blank, ANGLE, GAP, or FOIL in the identification column are accepted between SENARO and CURTIN cards. Cards with ANGLE, GAP, FOIL or SENARO are not accepted after the CURTIN card, except that a new SENARO card may follow BEGIN.

Either COUNT or TIMOPT cards are permitted between CURTIN and BEGIN, or between one BEGIN and the next, but not both. All TIMOPT cards must have the same value in the NL column between one BEGIN and the next. If NL = 1 there must be exactly 3 TIMOPT cards.

PS/6668
The data cards are read by a preliminary section of the programme which prepares a data file for the master section. All data cards containing a blank in the M column are just copied onto that file. Data cards with "1" in the M column are used to generate card images on that file which correspond to what would have been data cards for the same set of initial conditions, written with the simple option only.

There must be a single END card which must be the last one of the deck.

4.4. Description of output

Printed output on logical tape number 2 occurs immediately after each 50 lines (records) written onto the output tape and immediately after termination of a particular particle (JOBNUM) giving in the first case an asterisk at the left hand side of the page, and in the second one that JOBNUM and the content of the "record" counter NRCORD which counts the number of "records" written onto the output tape as described below. If the job is terminated by reading an END card in the data, a message is printed "PROGRAM ASTACK FINISHED AFTER ... MINUTES".

The intention of providing this printout is to make restarting as easy as possible. In fact, restarting at the JOBNUM following the last asterisk printed, at the "record" number NRCORD (in the N4 column of the SENERO card) ensures that all correct information on tape is conserved and that all possibly incorrect or incomplete information is overwritten whether the preceding run has been terminated normally or abnormally.
The output on magnetic tape with logical number 3 is grouped in binary records of 450 words. These records may be visualized as a matrix with 9 columns and 50 rows, each row corresponding to one "record" as used above. The information is written into this matrix row by row and transferred onto tape when the matrix is full.

The first rows are filled with information from the data card read from the expanded data file (logical number 5) from the SENARO to the first BEGIN, followed by a table of results, next information from the data file up to the next BEGIN, a table of results, etc.

Data cards are written into a row of the matrix in the following way:

IDENTIFICATION, N1, N2, VALUE, N3, N4, 0, 0, NRCORD. \( \text{(a)} \)

and the table of results is

BLANK, line number, step number, t, E, f, x, j, NRCORD \( \text{(b)} \)

where 0 is a word filled with zeros, and BLANK is a word filled with BCD blanks.

Thus, the first column of the matrix always contains an identification or a blank as is required for the ESTACK programme.

When the job is terminated by reaching the END card the remaining buffer area is filled with zeros and the buffer is written onto tape, followed by an end-of-file mark.
NOTES

a) The listing of the data card deck read from the input file (logical tape number 1) is printed out at the beginning of the output file (logical tape number 2) preceded by the message "LISTING OF DATA CARDS".

b) As said above, part or all the contents of the output tape can be copied onto the printed output tape. The heading gives the number of the first and of the last record which one wants to copy. To distinguish what type of row (a) or (b) we are reading, it is necessary that all the input cards between and SENARO and the next CURTIN with blanks in the IDENTIFICATION column have the N3 column left blank.

4.5. Essential information for the insertion of new subroutines

4.5.1. Arrangement of data

i) \( \text{CURRENT } (1,1) = \text{ present value of } E \)
\( \text{CURRENT } (2,1) = \text{ " " " } t \)
\( \text{CURRENT } (3,1) = \text{ " " " } x \)
\( \text{CURRENT } (4,1) = \text{ " " " } p \)
\( \text{CURRENT } (5, j + 1) = \text{ " " " } \phi_j \)
\( \text{CURRENT } (6, j + 1) = \text{ " " " } \Delta t_j \)

ii) The integer code for the IDENTIFICATION column is
\( \text{blank } = 1 \)
\( \text{ANGLE } = 2 \)
\( \text{GAP } = 3 \)
\( \text{FOIL } = 4 \)
\( \text{CURTIN } = 5 \)
iii) Let there be \( K \) cards following the \text{SENARO} card up to and including the \text{CURTIN} card. Then

\[
\text{IDENT}(j) = \text{integer code for IDENTIFICATION column}
\]

\[
\text{INDEX 1}(j) = N_1
\]

\[
\text{INDEX 2}(j) = N_2
\]

\[
\text{VALUE}(j) = \text{VALUE},
\]

as read from the \( j \)-th card, \( j = 1, 2, 3 \ldots K \).

iv) Further

\[
\text{NGAP} = \text{number of GAP being treated or about to be treated}
\]

\[
\text{FINOW} = \text{value of } j \text{ at the last gap treated}
\]

\[
\text{NSENIT} = \text{SENARO item selector}.
\]

4.5.2. Programme Structure

The programme is arranged as a main programme with three levels of subprogrammes as follows:

![Diagram of programme structure](attachment:image.png)
The fixed-point-variable NSENIT takes values 1, 2, 3, ... K during each simulated circuit of the accelerator. The function of subroutine STEP is to examine IDENT (NSENIT) and enter subroutines ANGLE, GAP or FOIL when its value is 2, 3 or 4 respectively. Once one of these subroutines has been entered, NSENIT may be used to select the appropriate values of NL, N2 and VALUE, VALUE (NSENIT + 1), VALUE (NSENIT + 2) .... etc. are the values of the subsequent parameters required for the calculation. When the calculation has been completed, the value of NSENIT is advanced by (n + 1), where n is the number of the cards which follow the ANGLE, GAP or FOIL card. In this way, when control is returned to subroutine STEP, NSENIT stands at the correct value for selection of NL and N2 from the next ANGLE, GAP or FOIL card.

To include a new formula for calculating a transformation at an ANGLE, GAP or FOIL, it is only necessary to write a few extra statements, and recompile the ANGLE, GAP or FOIL subroutine respectively.

4.5.3. The subroutines of the programme ASTACK

The programme is composed of the following subroutines :

S. UPDATE moves the output_tape (TAPE 3) to the "record", after which one wants to record the output data.

S. OUTPUT evaluates the phase POUT with respect to the gap NGPOUT from PNOW and calls.

S. BUFFOUT which writes the computed data (t, E, ϕ, ..., etc.), into the output buffer and, from time to time, onto the output tape.

S. BUFFEND fills the empty part of the last buffer with zeros and writes an end-of-file mark.
S. PREPAR reads the cards of the deck (input file TAPE 1) and makes conversion of the data either in simple or in compound form into a new series of data with simple option. The new data is recorded onto a file with logical number 5.

S. LOOP 1, S. LOOP 2, S. LOOP 3. LOOP i controls the passage of a particle through many revolutions in the accelerator when the i-th output option is chosen. These subroutines decide if the time t or the number of the steps is one of those for which output is required.

S. STEP Its function has already been explained.

S. ANGLE evaluates the time interval \( \Delta t \) to pass a circular section of the accelerator following a given law. It adds \( \Delta t \) to t and to every \( \Delta t_j \).

S. GAP calculates the voltage at the j-th gap at the time at which the particle crosses that gap. Calculates the new energy E and phase \( \phi_j \) values of the particle.

S. FOIL Although present in the deck, this subroutine is a dummy for the moment. It should calculate the change of E, p and x of the particle when this crosses a target.

S. GENMU calculates the frequency \( \nu \) of the voltage applied at the j-th cavity following a given law.

S. LUKUP identifies the IDENTIFICATION column and is used by the master section of the programme when it is reading from TAPE 3 or TAPE 5.

S. LUKIT has the same function as S. LUKUP but is used to read from the input file TAPE 1.

S. BUFFIN reads from the output tape.

S. LIST prints onto TAPE 2 part or all of the data recorded on the output tape when required.
4.5.3.1. Subroutine ANGLE - flow diagram

\[
\begin{align*}
J &= N1 = \text{INDEX 1 (NSENIT)} \\
K &= N2 = \text{INDEX 2 (NSENIT)} \\
\text{ALPHA} &= \alpha = \text{VALUE (NSENIT)}
\end{align*}
\] (I)

Go to (401, 402, 403, ...), \(J\) (II)

401
- Calculate \(\Delta t\) for \(N1 = 1\)
- Advance NSENIT

402
- Calculate \(\Delta t\) for \(N1 = 2\)
- Advance NSENIT

403
- Calculate \(\Delta t\) for \(N1 = 3\)
- Advance NSENIT (III)

Go to (450, 451, 452, ...), \(K\) (IV)

450
- \(x\) and \(p\) are not required
- \(N2 = 1\)

451
- Calculate \(x\) and \(p\) for \(N2 = 2\)
- Advance NSENIT

452
- Calculate \(x\) and \(p\) for \(N2 = 3\)
- Advance NSENIT

453 (etc.)

Add \(\Delta t\) to \(t\)
Add \(\Delta t\) to every \(\Delta t\)
RETURN (VI)

To include a new formula to compute \(\Delta t\), a new address must be included in (II) and a corresponding new section (III) must be written. Similarly, to include a formula for computing \(x\) and \(p\), a new address must be included in (IV) and a corresponding section (V) must be written.
4.5.3.2. Subroutine GAP - flow diagram

\[ J = N1 = \text{INDEX 1 (NSENIT)} \]
\[ K = N2 = \text{INDEX 2 (NSENIT)} \]
\[ \text{DOXP} = \text{VALUE (NSENIT)} \]  

(I)

GO TO (500, 501, 502, ...), J

(II)

500
Calculate \( V_j(t) \)
for \( N1 = 1 \)
Advance NSENIT

501
Calculate \( V_j(t) \)
for \( N1 = 2 \)
Advance NSENIT

502
etc.

(III)

550
GO TO (551, 552, 553, ...), K

(IV)

551
Calculate \( v_j(t = \frac{1}{2} \Delta t_j) \)
for \( N2 = 1 \)
Advance NSENIT

552
Calculate \( v_j(t = \frac{1}{2} \Delta t_j) \)
for \( N2 = 2 \)
Advance NSENIT

553
etc.

(V)

Calculate \( \phi_j \)
Calculate E
set \( t_j = 0 \)
set FNOW = \( \phi_j \)
NGAP = NGAP + 1

(VI)

Test DOXP

(VII)

zero
non-zero

(VIII)

Compute \( x \) and \( P \)
RETURN
To include a new formula to calculate $V_j(t)$, a new address
must be included in (II) and a corresponding new section (III) must
be written.

Similarly, to include a formula for computing $V_j(t - (1/2) \Delta t_j)$,
a new address must be included in (IV) and a corresponding section (V)
(S, GETNU) must be written.

4.6. Restrictions, files referenced, timing

The programme is subject to the following restrictions:

<table>
<thead>
<tr>
<th>Number of gaps</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of TIMOPT cards</td>
<td>500</td>
</tr>
<tr>
<td>Number of COUNT cards</td>
<td>500</td>
</tr>
</tbody>
</table>

The programme requires the following files:

- an input file with logical number 1,
- an output file with logical number 2,
- a file with logical number 3 for storing the results of the computation,
- a file with logical number 5 for the conversion of the input data
  into the expanded format.

A computation involving an ANGLE with NL = 1 and a GAP with
NL = 1 and N2 = 2 (see Appendix) was done at a rate of about
5000 steps/second on the CDC 6600 in CERN.
5. PROGRAMME BSTACK

5.1. Purpose

The output tape of the programme ASTACK contains the energies at a specific series of time instants of a set of particles circulating in an accelerator.

The accelerator is identified by its SENARO number, the time instants (particles), are identified by their line number (job number).

The stacking programme BSTACK will collect from this tape for a given SENARO number, the energies of a specified set of particles (JOENUM) at a specified series of line numbers (LINNUM). For each line number it will construct a histogram together with the associated means and variances representing the distribution of the selected particles in energy. The results are written onto magnetic tape for use by subsequent programmes.

5.2. Notation

M  number of line of output of ASTACK programme
B  number of particles being analysed
K  number of energy channels in the histogram
E_i  energy of the i-th particle
E  mean of energy distribution

\[
E = \frac{1}{B} \sum_{i=1}^{B} E_i
\]  

(5.1)

VE  variance of energy distribution

\[
VE = \frac{1}{B} \sum_{i=1}^{B} E_i^2 - (E)^2
\]  

(5.2)

A_i  number of particles with energies in channel i

PS/6668
average of the two energies defining upper and lower bounds of channel \( i \)

mean of grouped energies

\[
\bar{\varepsilon} = \frac{1}{B} \sum_{i=1}^{K} A_i \varepsilon_i
\]  

(5.3)

variance of grouped energies

\[
V_{\varepsilon} = \frac{1}{B} \sum_{i=1}^{K} A_i (\varepsilon_i^2 - (\bar{\varepsilon})^2)
\]  

(5.4)

number of channel containing largest energy

length of the histogram

d position indicator.

5.3. Method

The programme collects from magnetic tape the energy in the \( M \)-th line of output of each of a specified set of runs of ASTACK, runs being defined by their job numbers.

It produces a histogram based on \( K \) energy channels defined by \( (K + 1) \) input energies representing the boundaries of the channels, and calculates the mean and variance of both the actual distribution of energies and the grouped distribution, i.e. Eqs. (5.1), (5.2), (5.3) and (5.4).

When \( \overline{E} \) and \( \bar{\varepsilon} \), and \( V_E \) and \( V_{\varepsilon} \) are about equal, the energy spectrum of the particles is well represented by the histograms.

Histotrams are only output from the channel \( N \) containing the largest energy to the channel \( j \) containing the smallest energy

\[
L = N - j + 1
\]
The position indicator P is computed according to the formula:

\[ P = K_1 M + K_2 N + K_3 \]

where \( K_1, K_2, K_3 \) are integer constants given in the input data. Their choice is determined by the physical process under study.

Usually, in Swenson's method

\[ K_1 = 1 \]
\[ K_2 = -1 \]
\[ K_3 = \text{number of the reference channel which contains initially the particles (see paragraph 1.4).} \]

The position indicator P will be used by the CSTACK programme to shift the M-th column in the stacking matrix A in such a way that the element relative to the reference channel is placed on the principal diagonal, as explained in Section 1.

5.4. Input data

All data should be presented in a 6 column table with layout and formats as given in the following table containing all acceptable data cards:
<table>
<thead>
<tr>
<th>IDENT</th>
<th>N1</th>
<th>N2</th>
<th>VALUE</th>
<th>N3</th>
<th>N4</th>
<th>Row</th>
<th>col.</th>
<th>FORMAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 - 11</td>
<td>15 - 20</td>
<td>25 - 30</td>
<td>36 - 50</td>
<td>55 - 60</td>
<td>65 - 70</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A6</td>
<td>I6</td>
<td>I6</td>
<td>E 15.8</td>
<td>I6</td>
<td>I6</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SENARO</th>
<th>S</th>
<th>m</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOBNUM</td>
<td>J</td>
<td>ΔJ</td>
<td>2</td>
</tr>
<tr>
<td>JOBNUM</td>
<td>J_last</td>
<td>J_L</td>
<td>3</td>
</tr>
<tr>
<td>LINNUM</td>
<td>E</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>LINNUM</td>
<td>Δl</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>ENERGY</td>
<td>E_F</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>ENERGY</td>
<td>ΔE</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>BEGIN</td>
<td>K_1</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>K_2</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>K_3</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

The cards have the following effects:

Row 1 - S is the SENARO number of all jobs to be processed. m is the output indicators. If m = 131071 output will start at the beginning of the tape, if m ≠ 131071 output will start at the histograms with SENARO number m or at the END whatever occurs first. If m = 0 output starts overwriting the END.

NN3 = 0 output only onto tape
NN3 = 1 only printed output
NN3 = 2 output in the same time recorded onto tape and printed.
Row 2 - job number J is to be included in the processing;
Row 3 - all jobs with numbers in the range \( J_P(\Delta J)J_L \), inclusive, are to be included in the processing;
Row 4 - all jobs with SENARO number S on that portion of the tape from the beginning up to and including \( J_{\text{LAST}} \) are to be included in the processing. If no value of \( J_{\text{LAST}} \) is specified, all jobs with the relevant SENARO number will be included;
Rows 5, 6 - define the line numbers to be processed in exactly the same way as rows 2, 3 define the job numbers;
Row 7 - \( E \) is one of the boundary energy values delimiting the channels for the histograms;
Rows 8, 9 - \( E = E_P + j\Delta E \) (\( j = 0, 1, 2, \ldots, N_E \)) are a set of boundary energy values;
Row 10 - initiates the computation; \( K_1, K_2, K_3 \) are the constants used for computing \( P \).

There may be as many JOBNUM, LINNUM, ENERGY cards as necessary subject to:

a) JOBNUM cards with \( N_L = 2 \) should not occur if other JOBNUM cards are present,
b) an ENERGY card with \( N_L = 1 \) must be followed by an ENERGY card with \( N_L = 2, \)
c) the set of line numbers (energy bounds) generated by the LINNUM (ENERGY) cards must form a steadily increasing sequence.

Following the BEGIN card, another SENARO number can be specified by a new SENARO card followed by new JOBNUM, LINNUM and ENERGY cards. Each series of cards relative to a SENARO number must be terminated by a BEGIN card.
5.5. Output

Output on the logical tape no. 4 (NN3 = 0 or 2) consists of binary records of 512 words each. A buffer is filled as output becomes available and put onto tape each time when 512 words are reached. They contain, word by word:

SENARO

senaro number

ENERGY

lower channel bounds (K times)

JOBNUM

job numbers (3 times)

HISTOG

M
P
L

A_i (i = 1, 2, ..., L)

PARAM

E

VE

e

Ve

END

(once for every line number analysed)

The quantities A_i contained in the channels of the histogram are listed from the channel N to the channel (N - L + 1).

The printed output contains the same information in a different format.
5.6. Programme structure

The first part of the programme reads all the data from the input deck relative to a SENARO number, i.e. from the first SENARO card to and including the first BEGIN card.

When the SENARO card is read, the programme sets the position of the output tape no. 4 from which recording of the output should start (UPDATE). After reading the JOBNUM cards the vector JOBNUM (I) with B components which collects the job numbers to be included in the calculation is formed.

In the same way, after reading the LINNUM cards the vector LINNUM (I) with M components which is the series of the line numbers to be analysed is formed, and after reading the ENERGY cards the programme prepares the vector ENERGY (I) with \((K + 1)\) components which gives the boundary energies of the channels. The SENARO number and the boundaries of the energy channels are printed. During the second stage of the programme, which occurs after reading the BEGIN card, the vector EPSLON (I) is formed, which gives the steadily increasing series of the mean energies of the channels. These quantities serve to calculate \(\varepsilon\) and \(\bar{\varepsilon}\).

Then the main programme calls the SUBROUTINE SURCH. This finds on the input tape with logical no. 3, on which are recorded the trajectories of the particles (ASTACK), the job number I and the line number J which correspond to the elements of the vectors JOBNUM (I) and LINNUM (J), and copies the corresponding energy into the matrix element \(B(I,J)\).

At the same time a vector is formed:

\[
\text{JOBFND (I)}
\]
which is the series of the job numbers found on the tape and included in the calculation. When reading from the input tape is finished (either by reading an END record, or by reading a different SENARO record or when all JOBNUMS are found) the job numbers which were not found on the tape form the vector

JOBMIS (I).

Only when the last vector is empty the calculation is continued. The list of JOBEND (I) is output.

SUBROUTINE CMPUTE then follows. For each line number (M) it groups the quantities E(I, M) in the channels with boundaries given by the elements of the vector ENERGY. Then it calculates the position indicator P and the length L of the histogram. It evaluates, at last, the means and the variances (5.1), (5.2), (5.3) and (5.4). All these quantities will be output line by line.

When the first calculation is terminated, the programme returns to reading new data.

Further subroutines present in the programme are:

BUFFOUT, which writes the results onto the tape in the way mentioned in the previous chapter.
BUFFEND, fills the empty part of the last record with zeros and marks an end of file.
BUFFIN, reads buffer by buffer from the input tape with logical no. 3.
LUKUP, identifies the IDENTIFICATION column of the data on the tape no. 3 and on cards.
The flow diagram of the programme is

```
LUKUP
  MASTER PROGRAMME
  READ FIRST SERIES OF DATA
  PREPARE THE VECTORS
  JOENUM (I) LINNUM (I), ENERGY (I)
  LUKUP
  BUFFIN
  Surch
  E(I, J)
  JOBFND (I) and JOBMIS (I)
  COMPUTE
  construct histogram
  M, P, L
  means and variances
  BUFFEND
  read new data, if any, and
  restart the calculation.

UPDATE
  set the position
  of the output tape

BUFFOUT

IF JOBMIS (t) ≠ 0
  EXIT
```

5.7. Restrictions

- Number of particles (jobs) ≤ 150
- Number of lines (histograms) ≤ 160
- Number of energy channels ≤ 199

PS/6668
5.8. Files referenced
An input file with logical number no. 1
An output file with logical number no. 2
The input tape on which the results from the programme ASTACK are recorded, with logical number no. 3
The output tape onto which the histograms are to be recorded, with logical number no. 4

5.9. Timing
To process 160 lines of 120 jobs took 40 seconds CPU time on the CDC 6600 in CERN.

6. PROGRAMME CSTACK

6.1. General
In this section the programme CSTACK is described from the point of view of the conventions for the input and the output data.

The input for this programme is taken from the output tape of the BSTACK programme and/or from cards.

The procedure of the programme is suggested by the matrix method to compute the energy distribution of a stacked beam proposed by Swenson\(^5\), which has already been explained in Section 1.

The notation was already introduced in Section 5.

6.2. Procedure
The programme constructs a matrix A with K columns. The elements of the M-th column are the L numbers read from the M-th histogram on tape, normalized to unity, and placed such that the first number is in row P. The matrix obtained in this way may
then be modified according to information read from cards.

Two $K$ rowed vectors $V_A$ and $V_B$ are also read from cards.

The programme carries out a number of iterations on the matrix equation

$$V_{r+1} = A V'_r + V_B.$$ 

$V'_r$ is the vector $V_r$ shifted by $Q$ places such that its $i$-th component is

$$V'_r(i) = V_r(i + Q).$$

Output of $V_r$ in numerical form, accompanied by a graphical plot if required, may be requested in a flexible manner.

The matrix $A$ may be read entirely from cards.

After every cycle the programme adds the elements of the vector $V_{r+1}$ between the elements $K_2$ and $K_3 + \lambda r$ included; $\lambda$ is the constant already described in Section 2.

6.3. Specification of data

All data is specified in a 6 column table as follows:

<table>
<thead>
<tr>
<th>IDENTIFICATION</th>
<th>LABEL</th>
<th>VALUE</th>
<th>N1</th>
<th>N2</th>
<th>N3</th>
<th>COLUMNS</th>
<th>FORMAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 - 17</td>
<td>20 - 25</td>
<td>31 - 45</td>
<td>50 - 55</td>
<td>60 - 65</td>
<td>66 - 70</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2A6</td>
<td>16</td>
<td>E15.7</td>
<td>16</td>
<td>16</td>
<td>15</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The IDENTIFICATION column contains one of the words DETAIL, TAPE, HISTOGRAM, VECTOR, COUNT, BEGIN PLOT, or END, or may be blank.
The cards contain the following information:

**DETAIL**

- **NL** = 0, for stacking at the top
- **NL** ≠ 0, for stacking at the bottom
- **N2** = \( K \)
- **N3** = \( K_3 \)
- **VALUE** = \( \lambda \)
  - negative integer, for stacking at the bottom
  - positive (integer), for stacking at the top.

**PLOT**

- **NL** = \( C_1 \)
- **N2** = \( C_2 \)
  - parameters for graphical output
- **N3** = blank

**VALUE** = scale factor for graphical output, \( S \).

This card must be present only if the graphical output is requested.
In this case the elements \( v_j \) of \( V_r \) will be plotted, with \( j \) horizontal and \( v_j \) vertical. 51 positions are used in the vertical direction, the bottom position corresponding to \( v_j = 0 \) and the top position to \( v_j = S \). \( S \) should be not less than the greatest value expected in any element of any \( V_r \). 102 positions are used in the horizontal direction, the first one at the left hand side corresponding to the element \( j = C_1 + C_2 \) of the vector \( V_r \). \( C_2 \) defines the length of the horizontal direction. All the elements \( j = C_1 + iC_2 \) (\( i = 1, 2, \ldots, 100 \)) of the vector \( V_r \) will be plotted in this order.

**TAPE**

- **NL** = SEMARO number
- **N2** = blank
- **N3** = \( M \)
- **VALUE** = blank

PS/6668
If a card with identification TAPE is read, the matrix A will be constructed from K histograms, with SENARO number \( N_1 \), read from tape. The matrix may be adjusted from HISTOGRAM cards (see below). If a card with identification TAPE is not read, the matrix A must be constructed entirely from HISTOGRAM cards.

\( M_1 \) is the number of the first histogram to be read in.

**HISTOGRAM**

\[
\begin{align*}
N_1 & = M \\
N_2 & = P \\
N_3 & = L \\
\text{VALUE} & , \text{not used}
\end{align*}
\]

This card must be followed by L cards containing blanks in the identification column, and having numbers in the VALUE column. The L numbers so obtained are written into column M starting at row P of matrix A, replacing elements which may have been read from tape.

Other elements of column M are not altered. The new histogram must be normalized to unity.

**VECTOR**

\[
\begin{align*}
N_1 & = 1 \text{ for } V_A \\
N_2 & = 2 \text{ for } V_B \\
N_3 & = L \\
\text{VALUE} & = \text{output marker J}
\end{align*}
\]

PS/6668
This card must be followed by L cards containing blanks in the identification column, and having numbers in the VALUE column.

The L numbers so obtained are the P-th to (P + L - 1)-th elements of $V_A$ or $V_B$ (depending on NL).

The output marker J has the following significance:

- $J = -1.0$ do not output the vector
- $J = 0.0$ output the vector in numerical form
- $J = 1.0$ output the vector in numerical form together with a graphical plot.

```
COUNT
NL    = j
N2    = n_j
N3    = m_j
VALUE = J_j
```

A set of COUNT cards with $j = 1, 2, 3, ...$, causes the programme to perform $n_1$ cycles printing $V_r$ every $m_1$ cycles, then $n_2$ cycles, printing $V_r$ every $m_2$ cycles, etc. $J_j$ may take the value 0.0 or 1.0 and has the effect described above.

```
BEGIN
This card initiates the calculation. If the VALUE column is different from zero the programme uses the "expanded matrix", i.e., the first column is copied towards the left hand side of the matrix and the last column towards the right hand side until the length of the matrix is equal to the length of the vector $V_r$ before performing the product $A V_r$.
```

PS/6668
terminates the calculation.

There is also another card, SUPP, having a floating number α in the VALUE column. This card may appear after all DETAIL, TAPE, HISTOGRAM or VECTOR cards and before the BEGIN card. Its effect is to multiply the elements of the matrix A and of the vector \( V_B \) by the quantity \( 1 - \alpha \), and to add the quantity \( \alpha \) to the elements of the principal diagonal of A. This operation permits to compute the vector \( V_N \) for a stacking process with missing buckets, using the procedure discussed in reference 8.

6.4. Programme structure

The programme starts reading cards of the input deck. After reading of each card some operations follow.

After DETAIL card:
- the parameters \( \lambda, Q, K, K_3 \) are recorded.

After PLOT card:
- the scaling parameters S, C1 and C2 are recorded.

After TAPE card:
- the parameters NSEN and \( M_L \) are recorded. Then the programme calls the subroutine SEARCH which searches on the input tape with logical no. 3 (the output tape of BSTACK) that part containing the histograms with SENARO number NSEN. It collects all the histograms with \( M_L < M < M_L + K \) and construct the matrix MATRIX (I, M). The histogram no. M will become the matrix column no. \( M - M_L + 1 \). At the same time it calculates for the \( M \)-th histogram:
\( NFST \ (M) = P + 6 - M_\perp \quad \) (P read from the tape)
\( LENGTH \ (M) = L \quad \) (L read from the tape)
\( LAST \ (M) = P + 5 - M_\perp + L \).

At last is normalizes the columns of MATRIX to unity obtaining the new matrix \( A(I, M) \).

After HISTOGRAM cards:

- The programme modifies the matrix \( A(I, M) \) using information read from cards and as explained above.

After VECTOR cards:

- The matrix VECTOR \((I, NL)\) is filled by the elements read from cards. The vectors are then output in the requested form.

After COUNT cards:

- The vectors \( NCOUNT \ (1, j) \) \( NCOUNT \ (2, j) \), \( TYPE \ (j) \) are recorded.

After BEGIN cards:

- The programme starts the iterations.

The vector \( V_A \) is copied into the vector \( VNOW \ (I) \) using only the positions from \( I = 506 \) to \( I = 506 + K \) at the first step. Then the product between \( VNOW \) and \( A \) follows, the resulting vector being \( VCUR \ (I) \). The vector \( VVECTOR \ (I, 2) \) is added to \( VCUR \ (I) \) element by element and in such a way that the element \( I = 1 \) of \( VVECTOR \ (I, 2) \) corresponds to the element \( I = 506 \) of \( VCUR \ (I) \). The larger length of the vector \( VCUR \ (I) \) is useful in order not to lose the elements at the two sides when the distribution becomes very large after several cycles.
The next step is to shift all the elements of the vector VCUR (I) by Q places and to copy the vector VCUR into VNOW. At this point the product between VNOW (I) and A(I, M) is repeated. Each iteration is equivalent to a cycle of the stacking in the storage ring.

When required the expanded form of the matrix A(I, M) can be used in order that the number of the columns of A is equal to the number of the row of VNOW.

After a given number (r + 1) of iterations and in the way specified by the COUNT cards the vector VCUR (I) is output by means of the subroutine OUTPUT which, if required, calculates also the sum of the elements between \(K_3 + 500\) and \(K_3 + 500 + \lambda r\).

The flow diagram of the programme is the following:

The subroutines LUKUP and BUFFIN have the same functions as in the BSTACK programme. Subroutine SUPBUC is initiated by the card SUPP and makes the operations mentioned above.
6.5. Conventions for input data

All input data cards that are not blank in the IDENTIFICATION column are copied onto the output. A number may be written into the LABEL column of such cards, which will appear on the output, and serves to identify the vectors or histograms used.

The DETAIL card must be the first card of the deck, the END card the last one. There must be at least one COUNT card. There must be two VECTOR cards. If there is no TAPE card, there must be some HISTOGRAM cards.

After a BEGIN card a new DETAIL card may occur followed by a new series of cards relative to a new SENARO number which must be terminated always by a BEGIN card.

After completing the calculation, the computer will read the second set of data cards specifying the new calculation. All data for the first case will be carried forward to the new case except in so far as it is amended by new data cards.

There may be any number of TAPE cards in the deck.

All amendments to matrix A read from HISTOGRAM card are cumulative in their effect from case to case. On the other hand, if a new VECTOR card is read for a later case, the vector constructed for the previous case is completely abandoned.

6.6. Output

The output of the results of the r-th cycle, according to the data on the COUNT cards, consists of two parts:
i) output of the vector VCUR (I) in numerical form. We have a first line of the type "RESULT AFTER r CYCLES. SUM OF THE ELEMENTS (500 + K_3) TO (499 + K_3 + \lambda r) INCLUSIVE = .... REST = ....". When \lambda is an integer (and must be always an integer for stacking at the "bottom") the quantity REST is zero, but if \lambda is not an integer (as it may be for stacking at the "top") the "SUM" comprises only r* elements, with r* the biggest integer and smaller than r, and REST is (r - r*)times the contents of the (r + 1)-th element. For \lambda negative the summation of the elements is carried out between the elements (501 + K_3 + \lambda r) and (500 + K_3). The elements of the vector VCUR (I) follow line by line, 16 elements on a line. Printing starts at the first element bigger than 5 \times 10^{-5} and stops at the last element bigger than 5 \times 10^{-5}. The elements are output in three groups: the first one for I \leq 505 if there are some elements to be printed; the second one has all the elements from I = 506 to I = 506 + K, this group is always printed; finally, the third one for I > 506 + K if there is any element to be printed.

ii) On a new page, and only if requested, (output marker, J = 1.0), there is the graphical plot of the vector VCUR (I). Each unit of the horizontal axis corresponds in energy to the width of C_2 channels. The two values of the extremes of the horizontal scale are : (C_2 + C_1 + C_2) at the left hand which corresponds to the element I_1 = (505 + C_2 + C_1 + C_2) of VCUR (I) and C_2 + 102 C_2 at the right hand corresponding to the element I_2 = (505 + C_1 + 102 C_2). Generally, it is convenient to set C_2 = \lambda and C_1 = 0 for stacking at the top and C_1 sufficiently negative for stacking at the bottom.
6.7. Restrictions

\( C_1 \geq -500 \)

\( P \geq -505 \)

\( K \), number of a histogram, \( \leq 160 \)

\( L \), length of a histogram, \( \leq 100 \)

Number of COUNT cards \( \leq 20 \)

6.8. Files referenced

An input file with logical no. 1

An output file with logical no. 2.

A file with logical no. 3 containing the histograms, unless they are all defined by a histogram card.

6.9. Timing

To process a matrix A with 100 histograms and output results 45 times with plotting for a total of 150 cycles takes about 1 minute CP time on the CDC 6600 at CERN.
APPENDIX 1
Subroutines for ANGLE

In all ANGLE subroutines (except for NL = 3) \( a \) is the magnitude of the angle in units of \( 2\pi \) radians, given in the VALUE column of the ANGLE card. The available formulae for evaluating the time spent in the angle are:

\[ t = \frac{a}{a + b(E - E_1) + c(E - E_1)^2 + d(E - E_1)^2} \]

\( E_1, a, b, c, d \) should be written in this order into the VALUE column of cards following the ANGLE card.

\[ \text{NL = 2 } \quad t = (C_1 + C_2 \frac{E}{E_1^{1/2}} + C_3 E) a \]

\( C_1, C_2, C_3 \) are written in this order into the VALUE column of cards following the ANGLE card.

\[ \text{NL = 3 } \quad \text{causes the following calculations to be carried out:} \]

\[ \beta_{so} = \left( \frac{E_{so}^2 - E_{o}^2}{E_{s}^2} \right)^{1/2} \text{ on the first occasion that the ANGLE is met;} \]

\[ \beta_{so} = \beta_{sn} \text{ on subsequent occasions.} \]

\[ \begin{align*}
    t &= C_{so} \left( \frac{E_{so}^2 - E_{o}^2}{E_{s}^2} \right)^{1/2} \\
    \beta_{sn} &= \left[ 1 - \left( \frac{1 - \beta_{so}}{E_{o}^2 + K_{so} E_{so}^2} \right)^{-1/2} \right]^{1/2}
\end{align*} \]

PS/6668
E_s must be written in column 4 of the ANGLE card, followed by E_o, C and K in the next three rows of column 4. An ANGLE for which NL = 3, if used, must come first in the SEBARO data.

\[
NL = 4 \quad t = \alpha \gamma \exp \left[ B \log \frac{\gamma_i^2 - 1}{\gamma^2 - 1} \right]
\]

with \( \gamma = E/E_o \) and \( E_o = 0.5109 \) MeV. The energy E should be expressed in MeV. \( \gamma_i, \alpha, B \) are given in this order in the VALUE column of the cards following the ANGLE card. This routine is useful for the FTAG accelerator.

\[
NL = 5 \quad t = \frac{\alpha}{a_1 \delta^5 + a_2 \delta^4 + a_3 \delta^3 + a_4 \delta^2 + a_3 \delta + a_2}
\]

with \( \delta = E - E_1 \).

\( E_1, a_2, a_3, a_4, a_5, a_6, a_7 \) are given in this order in the VALUE column of the cards following the ANGLE card.

\[
NL = 6 \quad t = \frac{\alpha}{\int_0^1 \frac{1 + 2 H_1(E - E_o)}{E - E_o}}
\]

\( E_1, f_o \) and \( H_1 \) are given in this order in the VALUE column of cards following the ANGLE card.

This routine corresponds to Eq. (2.4) for a bucket with area and \( \Gamma \) constant.

\( N2 = 1 \) This value of N2 means "betatron oscillations not required". \( x \) and \( p \) are unaltered. All other values of N2 will cause an error exit.
APPENDIX 2
Subroutines for GAP

In all GAP subroutines (except for NL = 4) h is the harmonic number of the cavity and is always given in the VALUE column of the last of the cards following the GAP card.

The available formulae for evaluating V and v are:

\[ V(t) = V_o = \text{constant} \]
Write \( V_o \) into the VALUE column following the GAP card.

\[ V(t) = a + b(t - t_1) + c(t - t_1)^2 \]
Write \( t_1, a, b, c \) in this order into the VALUE column of cards following the GAP card.

\[ V = a \beta_{so} \]
where \( \beta_{so} \) has the value obtained at the previous ANGLE with NL = 3.
Write a into the VALUE column following the GAP card.

\[ V = a \beta_{so} \]

With this value of NL none of the calculations associated with GAPs described in the paragraph 4.3.1. of Section 4 are carried out. Instead NL = 4 simply causes
\[ C (1 - \beta_{sn}^2)^{-1/2} \]
to be added to the particle energy, where \( \beta_{sn} \) has the value obtained at the previous angle for which NL = 3.

Following the GAP card, write C in the next row into the VALUE column.
NL = 5  yields a modulated voltage \( V(t) \) according to the following law:

\[
\begin{align*}
\text{For } & \quad t \leq t_2 \\
V(t) &= V_0 \exp(t/\tau) \\
\text{and for } & \quad t \geq t_2 \\
V(t) &= V_1 = \text{constant } (= V_0 \exp\left[\frac{t_2}{\tau}\right])
\end{align*}
\]

Following the GAP card write \( V_0, V_1, \tau, t_1, \) and \( t_2 \) in this order into the VALUE column of the next five rows. \( t_2 \) is a parameter for the associated frequency modulation \( \nu(t) \) with \( N2 = 4 \).

\[ N2 = 1 \quad \nu(t) = \nu_0 = \text{constant} \]
Write \( \nu_0 \) and \( h \) into the VALUE column of the cards following the voltage parameter cards.

\[ N2 = 2 \quad \nu(t) = a + bt \]
Write \( a, b \) and \( h \) into the VALUE column of cards following the voltage parameter cards.

\[ N2 = 3 \quad \nu(t) = a + bt + ct^2 \]
Write \( a, b, c \) and \( h \) into the VALUE column of cards following the voltage parameter cards.

\[ N2 = 4 \]
represents a compound frequency modulation \( \nu(t) \) to be used the voltage modulation. \( NL = 5 \).

\[
\begin{align*}
\text{For } & \quad t \leq t_1 \\
\nu(t) &= \nu_0 = \text{constant}.
\end{align*}
\]

\[
\begin{align*}
\text{For } & \quad t_1 \leq t \leq t_2 \\
\nu(t) &= \nu_0 + a(t - t_1)^2
\end{align*}
\]
And for \( t \leq t_2 \)
\[ v(t) = v_1 + a(t - t_2). \]
(Usually \( v_1 = v_0 + a(t_2 - t_1)^2 \)).

Write \( v_0, v_1, a, v \) in this order in the VALUE column of cards following the voltage parameter cards with \( N_1 = 5 \). Two cards containing all blanks must be added. Write \( h \) in the VALUE column of the next card.

**Subroutines for "missing buckets" procedure**

This is given by \( N_1 = 6 \) and \( N_2 = 4 \). The parameters for \( N_1 = 6 \) are exactly the same as for \( N_1 = 5 \) and must be given in the same order and in the same way.

The only difference is that the VALUE column of the two cards, which were left blank for the sequence \( N_1 = 5, N_2 = 4 \), are now filled in the order by the values of two parameters \( A_1, A_2 \) which are numbers with \( 0 < A_1, A_2 < 1 \).

The subroutine \( N_1 = 6 \) has the effect of setting \( V = 0 \) if the new phase \( \phi \) of the particle is in the interval

\[ A_1 < \phi < A_2 \]

otherwise \( V \) is evaluated as in the case \( N_1 = 5 \).
APPENDIX 3
Example of compound data option

Suppose we want to study particles with 5 initial energies $E_1 = 2, 3, 4, 5, 6$ in a device with 2 gaps under three initial phase conditions:

phase gap 1 $\phi_1 = 0.2$, phase gap 2 $\phi_2 = 0.0$
$\phi_1 = 0.4$ $\phi_2 = 0.3$
$\phi_1 = 0.6$ $\phi_2 = 1.0$

We allocate 1001 as the first job number. Then $N_G = 2$, $N_E = 5$, $N_F = 3$, $N_J = 1001$ and the data are written as follows:

<table>
<thead>
<tr>
<th>Job</th>
<th>Job</th>
<th>Energy</th>
<th>Gap</th>
<th>Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2.0</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0.2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td></td>
<td>1001</td>
<td></td>
</tr>
</tbody>
</table>

The effect of these data is exactly the same as that of the following data written with the simple option only:

<table>
<thead>
<tr>
<th>Job</th>
<th>Job</th>
<th>Energy</th>
<th>Gap</th>
<th>Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1001</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JOE</td>
<td>NUM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENERGY</td>
<td>2.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FIGAP</td>
<td>1</td>
<td>0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FIGAP</td>
<td>2</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEGIN</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JOE</td>
<td>NUM</td>
<td>1002</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENERGY</td>
<td>3.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FIGAP</td>
<td>1</td>
<td>0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FIGAP</td>
<td>2</td>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JOBNUM</td>
<td>ENERGY</td>
<td>FIGAP</td>
<td>FIGAP</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>-------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>1006</td>
<td>2.0</td>
<td>0.4</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>1007</td>
<td>3.0</td>
<td>0.4</td>
<td>0.3</td>
<td></td>
</tr>
</tbody>
</table>

PS/6668
APPENDIX 4

Subroutine OTHER

The following sequence of cards is also accepted by the ASTACK programme. It is used when the input data for the initial conditions of particles and gaps are computed for an advanced time $t$ during the acceleration cycle in order to reduce the computer time (see paragraph 2.10 of Section 2).

The data should be given in the following sequence:

<table>
<thead>
<tr>
<th>Row</th>
<th>M</th>
<th>IDENT</th>
<th>N1</th>
<th>N2</th>
<th>VALUE</th>
<th>N3</th>
<th>N4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>OTHER</td>
<td>4</td>
<td></td>
<td>$t$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>$\phi_s$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>a</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>b</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td>$E_0$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td>$E_1$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td>$DE$</td>
<td>$N_p$</td>
<td>ID</td>
<td></td>
</tr>
</tbody>
</table>

Some symbols are defined in the formula (2.33) of Section 2, and the remainder is given by:

$$a = \sqrt{2\hbar}$$

$$b = \frac{1}{r_s} (f \frac{df}{dE})$$

$$E = E_1 + (n-1) DE, \; n = 1, 2, \ldots, N_p.$$

The first job number (JOBNUM) is $ID + 1$; and the job number $JOBNUM = ID + n$ is associated with a particle with energy $[E_1 + (n-1) DE]$. 

PS/6668
REFERENCES


6. E. Keil and A. Nakach, "Beam stacking at high values of \( \Gamma = \sin \phi \)", CERN 66-9.


9. The original programmes were written by J.S. Hornsby and are described in the following internal CERN documents:
   a) A programme for accelerator studies (Progr. 02004/A), Ref. 982/p, 28.3.1961.
   b) Addendum to a programme for accelerator studies (Progr. 02004/A), Ref. 4083/p, 6.6.1962.
c) Revised input facilities for programme 02004/A, DD/CO, Ref. 3633/p, 2.4.1962.

d) Ancillary programme 1 for processing the results of a programme for accelerator studies (programme 02004/B), Ref. 983/p, 30.3.1961.


10. CERN computer, 6000 series
   a) SCOPE general reference
   b) FORTRAN

   CERN - DD Division 1968.
