

ビーム光学総合計算コード SAD*

COMPUTER PROGRAM COMPLEX SAD FOR ACCELERATOR DESIGN, SIMULATION AND COMMISSIONING

大西幸喜^{†A)}、森田昭夫^{A)}、杉本寛^{A)}、小磯晴代^{A)}、周徳民^{A)}、大見和史^{A)}、生出勝宣^{B)}

Yukiyoshi Ohnishi^{†A)}, Akio Morita^{A)}, Hiroshi Sugimoto^{A)}, Haruyo Koiso^{A)}, Demin Zhou^{A)},

Kazuhiro Ohmi^{A)}, and Katsunobu Oide^{B)}

^{A)}High Energy Accelerator Research Organization (KEK)

^{B)}European Organization for Nuclear Research (CERN)

Abstract

A computer code, *SAD* has been developed at KEK since 1986 in order to accomplish optics design, particle-tracking simulations, and operations of accelerator beams with the EPICS system. *SAD* is applied for a number of accelerators including TRISTAN, KEK-ATF, SLAC-FFTB, KEKB, KEK-PF, J-PARC, BEPC-II, and SuperKEKB. It has been also used for design works of the future machines such as ILC, CEPC, FCC-ee, and muon colliders. We present a review of computer code *SAD* with respect to a role of computer code and experiences on using *SAD* at various accelerator projects.

1. はじめに

SAD [1] は、1986 年頃から KEK で開発されてきた加速器設計のための汎用計算プログラムであり、Strategic Accelerator Design の略称である。1985 年の OHO セミナーが開催されたおり、木村嘉孝名誉教授(当時のトリスタン計画推進部電子リング研究系主幹)の開会あいさつで「KEK 独自の計算機コードが存在しないのが残念である。」と述べられ *SAD* 開発のきっかけとなったことが、第 1 回 *SAD* ワークショップ講演記録集 [2] の冒頭に記述されている。*SAD* を使用する前は、MAGIC(挿入部の光学設計、マッチング)、HARMON(色収差補正)、PATRICIA(粒子トラッキング)など、別々のプログラムを目的に合わせて使い分けていた。したがって、作者が異なるために、使用している物理モデルや近似がそれぞれ異なるという不満があった。*SAD* では、こうした機能を一つにまとめた計算コードとなっている。

もともと、*SAD* は加速器設計のために作られた計算コードであったが、現在では、様々な機能が必要に応じて追加され、加速器の設計から加速器の制御まで多岐に渡って使用できるようになっている。実際の加速器で、ステアリングを使って得られた軌道の変化から応答行列を測定し、Twiss パラメータなどの物量を推定しモデルと整合するように磁石の磁場補正量を計算することが、オンラインでできる。そして、すぐさま補正結果を確認でき、また再帰的に補正が可能である。オフラインで、一週間後に答えが出て、もはや、その結果は適用できるかどうか疑わしい。オンラインで、「ビームを用いた光学系の測定」→「補正量の計算」→「実際に電磁石を設定して補正」といった全ての過程を一度に行うことができる *SAD* のようなオールインワンの加速器汎用計算コードが世の中に、ほとんどないのが現実である。

SAD がこのような発展をしてきたのは、「必要とする人が、必要な機能を実装してきた」ことに由来する。日本で、TRISTAN、KEKB、SuperKEKB、J-PARC、Linear Collider といった比較的大きな加速器プロジェクトが続けられたことも、*SAD* を発展させてきた大きな原動力であったとも言える。

さて、*SAD* が全てについて優れており、短所がないというわけではない。スピン偏極やテラレーマッパは実装可能であるが、現在簡単に取り扱えるようになっていない。幾何学的配置(GEO)や Twiss パラメータ等の計算(OPTICS)およびマッチングを司るコード(FFS)、ビームマトリクス法(EMIT)、粒子トラッキング(TRACK)が一つのパッケージとして組み込まれているが、基本的に別々の計算ルーチンであるため、それらの間で写像が整合しているかどうかの確認が実は困難である。したがって、前述した *SAD* 以前のコードがひとつにまとまっていないという不満な点が、完全に解消できているわけではない。また、マニュアル、ユーザーガイド、レファレンスなどが貧弱である(ほとんど皆無)。使ってみようと思っている人に気軽にソースを配布できるようになっていない、サポート環境が整っていないなど、一般ユーザーに優しいとは言えない。

しかしながら、加速器計算コードの必要性は自明である。海外の加速器プロジェクトにおいても加速器計算コードの開発および保守には、様々な苦勞があると思われる。世界で使用されている加速器計算コードを以下に挙げる。

- Accelerator Toolbox (AT) [3]
- Bmad with PTC [4]
- Elegant [5]
- MAD-X with PTC [6]
- PTC [7]
- SAD
- SixTrack [8]

* 本研究は JSPS 科研費 17K05475 の助成を受けたものです。

† yukiyoshi.ohnishi@kek.jp

想定される加速器モデルと背景とする物理が、それ

それぞれ異なるため、使用には一長一短がある。これらの計算コードにおいて、何を比較すべきかは簡単ではない。

2. SAD の特徴

SAD の実装は、FORTRAN 言語と C 言語である。1994 年頃に、本格的なスクリプト言語 (SADScript) インタープリターが実装され、汎用プログラミング環境が整った。SADScript インタープリターは、Wolfram Mathematica をモデルとしている。完全互換ではないが、三角関数、指数関数、Bessel 関数などの数学関数や文字列操作およびリスト操作ができるように豊富な組み込み関数が用意されている。また、対話的に計算ができることは、多くの場合で便利である。現在では、ランタイムでの拡張や他言語での関数呼び出しもサポートしている。

グラフィクスを含むユーザー・インターフェースとして、Tk widget toolkit をラップした Tkinter が用意されている [9]。これにより、汎用なグラフ表示や加速器に特化したビームラインに沿った光学パラメータのグラフ表示などが簡単に行えるようになっている。

File I/O および Network I/O についても、SADScript の組み込み関数としてサポートされている。また、EPICS Channel Access も用意されている。実際の加速器の運転情報取得および、そのリアルタイム表示や加速器制御も EPICS システムを利用して行うことができる。

2.1 加速器ビームラインのモデリング

SAD で定義される加速器ビームラインを構成する主要要素を、Table 1 に示す。磁石要素は、ROTATE というパラメータでビーム軸 (進行方向) のまわりで回転させることができ、スキューモードの磁場を表現できるようになっている。磁石の回転の向き、ROTATE の符号は、後に述べられる座標軸、 $\vec{e}_x \times \vec{e}_y = -\vec{e}_z$ で定義される。

Table 1: List of SAD Elements

element	symbol
ドリフト空間	DRIFT
偏向磁石	BEND
4 極磁石	QUAD
6 極磁石	SEXT
8 極磁石	OCT
加速空洞	CAVI, TCAVI
磁石、加速空洞の複合体	MULT
ソレノイド磁場の境界	SOL
ビームビーム相互作用	BEAMBEAM
汎用の写像	MAP
物理口径	APERT
ビーム光学系の基準点	MARK
要素の並び	LINE

あらかじめ組み込まれている写像は、ハミルトニアンに基づいており、シンプレクティック条件を満たすようになっている。ドリフト空間のハミルトニアンは、

$$H = -\sqrt{p^2 - p_x^2 - p_y^2} + \frac{E}{v_0} \quad (1)$$

である。SAD における p_x および p_y は、ハミルトニアンの定義により軌道の傾きではなく正準運動量である。ドリフト空間の写像を求めると、

$$x_2 = x_1 + L \frac{p_{x1}}{\sqrt{p^2 - p_{x1}^2 - p_{y1}^2}} \simeq x_0 + L \frac{p_{x1}}{p} \quad (2)$$

である (L はドリフト空間の長さ)。これから変位は運動量に対して非線形であることがわかる (色収差が発生する原因)。これを解ける形に近似した後のドリフト空間の転送行列は、よく知られており、 $M^T J M = J$ を満たすシンプレクティックな行列である。SAD では、ハミルトニアン力学の枠組の中で計算が行われる。実際には、このように何らかの近似が必要となる場合が多い。その手法には無限のやりかたがあるはずであるが、SAD では物理的に意味のある、ハミルトニアンに基づく近似法を使用している。

磁石の磁場は、基本的に次のように表される。

$$K_n = \frac{B^{(n)} L_s}{B\rho} \quad (n = 0, 1, 2, 3, \dots) \quad (3)$$

$$B\rho \equiv p_0/e \quad (4)$$

K_n は磁石の積分磁場であり、 L_s は、磁石の有効長 (直線) である。正の K_n は水平方向に収束能力を持つことを意味する。このような K 値の表現を用いている理由は、厚みのある磁石を取り扱うことに意味があるという思想が SAD にあるからである。

4 極磁石の K_1 を考えた場合、転送行列 ($K_1 > 0$) は近似的に、

$$M = \begin{pmatrix} \cos \sqrt{K_1 L_s} & \sqrt{\frac{L_s}{K_1}} \sin \sqrt{K_1 L_s} \\ -\sqrt{\frac{K_1}{L_s}} \sin \sqrt{K_1 L_s} & \cos \sqrt{K_1 L_s} \end{pmatrix}$$

となる。大抵の場合、厚み L_s は焦点距離 $1/K_1$ よりも非常に小さく、

$$M \simeq \begin{pmatrix} 1 & 0 \\ -K_1 & 1 \end{pmatrix} + (K_1 L_s) \begin{pmatrix} -1/2 & 1/K_1 \\ K_1/6 & -1/2 \end{pmatrix} + \dots \quad (5)$$

のように近似できる。このように、 M の最低次を K_1 のみで表現できることは微分形である $B'/B\rho$ と比較して有意義である。

SAD では、MOMENTUM、MASS、CHARGE などといった大域的なパラメータが用意されている。MASS のデフォルト設定は、電子質量 (eV)、CHARGE は +1 である。SAD で用いられる単位系は MKS である。

2.2 座標系

大局的な座標上に加速器ビームラインの構成要素が配置されるが、基本的にドリフト空間と偏向磁石によって基準軌道は決定される。大局的座標は、 $(G_X, G_Y, G_Z, \chi_1, \chi_2, \chi_3)$ という6個のパラメータで記述される。ビームラインの先頭と後述する局所的座標の関係は、初期設定では、 $(G_X, G_Y, G_Z) = (s, -x, -y)$ となっている。また、局所的座標の回転は、 (χ_1, χ_2, χ_3) の3つの角度で表現される。偏向磁石のエレメント (BEND) は、ANGLE と呼ばれるパラメータを指定することによって局所的座標を回転させる (Fig. 1)。ANGLE は、

$$ANGLE = \frac{BL_a}{B\rho} \quad (6)$$

と表され、 L_a は磁石の有効長 (弧) であり ANGLE に依存する。

基準軌道は、基準粒子の軌道であるが、実際に粒子が通る軌道でなくてもかまわない。世界点を (x, y, z, t) としたときに、直行座標系では、時刻 t をパラメータとして物理座標を (x, y, z) と選ぶのが普通であるが、加速器の場合、基準軌道上に沿った位置を示す s をパラメータとして (x, y, t) を物理座標とする方が大抵の場合便利である。 z は相対的な位置を示すが、 $z = s - vt = -v(t - t_0)$ なので時間と捉えるべきものである (早く到達した粒子は +、遅く到達した粒子は -)。ここで使用する局所的座標系は、右手系である。基準軌道の狭い周辺領域を通過する粒子の座標は、基準軌道上の各点 $\vec{r}(s)$ を使って

$$\vec{u}(s) = \vec{r}(s) + xe_x^-(s) + ye_y^-(s) \quad (7)$$

と表せる。ただし、 x 軸および y 軸は「加速器ハンドブック」に記述されているように決める [10]。この決め方では、基準座標のある1点で x 軸と y 軸を決めると、全ての点で x 軸と y 軸が決まる。注意すべき点は、各点での主法線の向きを x 軸とする Frenet-Serret 座標を使って x 軸と y 軸を決めていない点である。

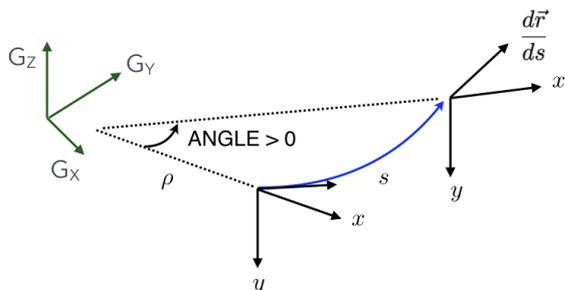


Figure 1: Definition of ANGLE.

2.3 FFS におけるマッチング

FFS では、軌道と写像の計算のために6次元のシンプレクティックなトラッキングが使われるが、光学関数を求める場合は、4行×5列 (4次元と分散) に

縮小された転送行列が用いられる (k64 バージョンでは6次元もサポートしている)。ビームラインの最初の要素は、軌道や Twiss パラメータの初期値を格納する MARK 要素でなければならない。計算方法を指定するフラッグを、Table 2 に示す。周期条件を満たすように解く CELL や、加速要素があってもビームラインの始点と終点で設計運動量を一定に保つ RING がある (加速位相は自動調整される)。TRPT を指定した場合、ビームラインはリナックもしくはビーム輸走路として取り扱われ、加速によって設計運動量は変化する。

Table 2: Flags in FFS

Flag	Antonym	Default	Effect
CELL	INS	INS	Periodic condition
RING	TRPT	RING	Momentum is constant.

一般的に、転送行列や Twiss パラメータは、磁石の磁場の強さや有効長の非線形な関数となっている。転送行列の要素や Twiss パラメータを希望の値に合わせる場合、磁石の磁場や有効長を解析的に求めることはできない。これらの問題を解く場合、非線形連立方程式の解として与えられ、数値解として求めるしかない。このように転送行列の要素や Twiss パラメータなどを希望する値に合わせる計算手法をマッチングと呼ぶ。

非線形連立方程式の数値解を求める場合、SAD では多次元ニュートン＝ラフソン法と最急降下法を併用している。多次元ニュートン法の解法においては、特異値分解 (SVD) を使用している。ニュートン法は、古典的であるが単純であり強力である。SAD の FFS に組み込まれているマッチングのアルゴリズムを Fig. 2 に示す。

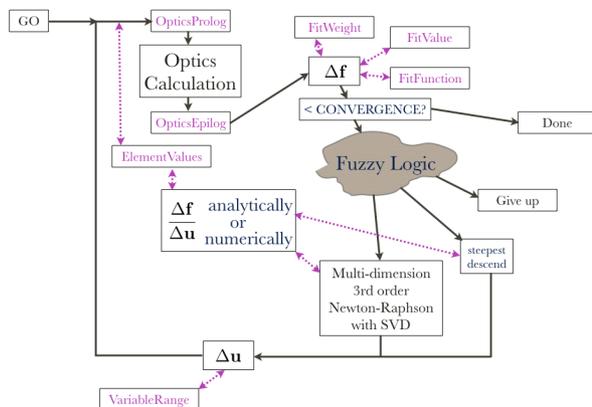


Figure 2: Matching procedure of SAD/FFS. FitValue and FitFunction are also shown.

マッチングの対象となる組み込まれている光学関数を Table 3 に示す。物理座標における分散には、physical と示されている。これらは、基準座標 (非結合) と物理座標を関係づける X-Y 結合行列 ($r_1 \sim r_4$ の要素を持つ 4×4 行列) によって関係づけられる。

Table 3: Optical Functions for Matching

Variable	Symbol	Unit
AX, AY	α_x, α_y	
BX, BY	β_x, β_y	m
NX, NY	ψ_x, ψ_y	1/2 π
EX, EY	η_x, η_y	m
EPX, EPY	η_{px}, η_{py}	
R1, R4	r_1, r_4	
R2	r_2	m
R3	r_3	1/m
PEX, PEY	η_x, η_y (physical)	m
PEPX, PEPY	η_{px}, η_{py} (physical)	
DX, DY	$\Delta x, \Delta y$	m
DPX, DPY	$\Delta p_x, \Delta p_y$	

4 極磁石の K_1 による Twiss パラメータ、分散、X-Y 結合などの変化は微分による解析的に解が求まるものがある (ただし、4 極磁石に軌道オフセットがない場合)。SAD では、可能ならば解析的な解を用いて、それ以外は数値微分を用いている。解析解を用いることが可能な場合でも、ユーザーが選択的に数値微分を用いることができる。光学関数だけでなく、 G_X 、 G_Y 、 G_Z 、 χ_1 、 χ_2 、 χ_3 などの幾何関数もマッチング可能である。しかも、FitValue や FitFunction を介して、目標関数の区間拘束やエミッタンスといった任意の目標関数をマッチングに追加することができる。

2.4 ビームマトリックス

EMIT コマンドは、平衡ビームマトリックスを計算する。計算するビームラインは、陽電子のリングとなる。平衡状態におけるビームマトリックス Σ は、6×6 行列で

$$\Sigma = M \Sigma M^T + \Lambda \quad (8)$$

$$\Sigma \equiv \langle q_i q_j \rangle, \quad q_i = (x, p_x, y, p_y, z, \delta \equiv \Delta p/p_0)$$

を解くことによって得られる。ここで、 M は放射減衰を含むリング 1 周の転送行列であり、 Λ はシンクロトロン放射とイントラビーム散乱による拡散行列である。 M が 3 つ (x, y, z) の自由度間結合を含んでいる場合においても、ビームマトリックス法は適用可能である。平衡エミッタンスは、 $J\Sigma$ の固有値の絶対値として各自由度ごとに得られる。ここで、 J は第 2.1 章で記述されているシンプレクティック行列である。同じ結果を返す関数、Emittance[] も用意されている。こちらは、オプションを指定することによって、より多くの情報をリスト形式で得られる。EMIT によって正しく計算するためには、FSHIFT、PBUNCH、MINCOUP などが適切に設定されている必要がある¹。計算方法を指定するフラッグを、Table 4 に示す。

¹ INTRA を指定した場合、PBUNCH と MINCOUP が必要。

Table 4: Flags in Emittance Calculation

Flag	Default	Effect
COD	COD	Calculate COD
RFSW	RFSW	Acceleration by CAVII
RADCOD	NORADCOD	Synchro. rad. energy loss
INTRA	NOINTRA	Intra-beam scattering

2.5 粒子トラッキング

力学口径を求める場合、解析的に計算することは困難である。したがって、力学口径を求めるためには、粒子トラッキングに頼らざるを得ない。また、多粒子系の複雑な相互作用がある場合においても、粒子トラッキングの手法は使われる。

SAD では、6 次元座標 $(x, p_x, y, p_y, z, \delta)$ を粒子トラッキングで用いる。これらは、全て正準変数である。組み込みで用意されている要素の写像には、厚みを考慮して考えられる物理が盛り込まれている。

粒子トラッキングを計算するための関数、Track-Particles[] が用意されている。トラッキングを行うビームラインの始点と終点を指定し、始点における粒子分布と生存フラッグを用意すると、終点における粒子分布と生存フラッグが返り値として計算される。なお、粒子分布は、それぞれの座標についてリストであり、生存フラッグは 1 もしくは 0 のリストである。Table 5 に、粒子トラッキングで用いられるフラッグを示す。

Table 5: Flags in Particle-tracking

Flag	Antonym	Effect
RING	TRPT	Momentum constant
RFSW	NORFSW	Acceleration by CAVI
RAD	NORAD	Synchrotron rad. energy loss
FLUC	NOFLUC	Diffusion due to synchrotron rad.

3. SAD の実例

SAD を対話的に起動する場合、以下のように (パスが /SAD/bin/ の場合) するとプロンプトが返ってくる。

```
$ /SAD/bin/gs FFS
~ 省略 ~
In [1]:=
```

gs は、"go sad" の略称であり、"ghostscript" と混同しないように注意が必要である。また、SADScript で書かれた実行スクリプト (台本と呼ばれる) を実行させる場合、

```
$ /SAD/bin/gs fodo.sad
```

とする。fodo.sad のソースコードを以下に示す。

```
OFF CTIME; ON ECHO;
;
DRIFT L1 =(L =0.2 ) L2 =(L =1.5 )
```

