

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4368377号
(P4368377)

(45) 発行日 平成21年11月18日(2009.11.18)

(24) 登録日 平成21年9月4日(2009.9.4)

(51) Int.Cl. F I
F 4 1 G 3/00 (2006.01) F 4 1 G 3/00

請求項の数 13 (全 15 頁)

(21) 出願番号	特願2006-507941 (P2006-507941)	(73) 特許権者	500288164
(86) (22) 出願日	平成16年3月4日(2004.3.4)		トータルフェルスバレッツ フォルスクニ
(65) 公表番号	特表2006-519358 (P2006-519358A)		ングスインスティテュート
(43) 公表日	平成18年8月24日(2006.8.24)		スウェーデン王国、エス-164 90
(86) 国際出願番号	PCT/SE2004/000309		ストックホルム
(87) 国際公開番号	W02004/079289	(74) 代理人	110000475
(87) 国際公開日	平成16年9月16日(2004.9.16)		特許業務法人みのり特許事務所
審査請求日	平成19年3月2日(2007.3.2)	(74) 代理人	100068032
(31) 優先権主張番号	0300560-0		弁理士 武石 靖彦
(32) 優先日	平成15年3月4日(2003.3.4)	(74) 代理人	100080333
(33) 優先権主張国	スウェーデン(SE)		弁理士 村田 紀子
		(74) 代理人	100115222
			弁理士 徳岡 修二
		(74) 代理人	100124796
			弁理士 重本 博充

最終頁に続く

(54) 【発明の名称】 軌道上の発射物を所望の地点で計算された時間に作用させる方法

(57) 【特許請求の範囲】

【請求項1】

発射物を所望の地点で作用させるように、該発射物の想定し得る2つの仰角及び関連づけられた飛行時間を準実時間で計算する方法であって、

前記発射物の発射方向がある垂直面(XZ平面)の方位角が、従来技術による方法、例えば、該発射物が作用すべきターゲットの方向を直接測定することによって決定され、

原点が前記発射物の発射地点に固定され、X軸が水平面と平行に固定され、

前記仰角及び飛行時間は、計算部及び論理部の2つの主要部に区分される処理で計算され、

該計算部は、前記発射物の直径(d)、質量(m)、空気抵抗係数(C_d)、及び発射速度(V_{launch})から始めて、軌道上の発射物の位置及び関連付けられた飛行時間を離散的に適時に計算し、

論理部は、該論理部での最大誤差(acc)、所望の高さの下限値(lh)、前記ターゲットまでの水平距離(x_p)、および該ターゲットに対する相対高さ(z_p)から始めて、

第1の仰角方向(θ_{launch})を設定し、

発射物の位置および飛行時間の計算を監視し、

前記計算を中断し、ここで、前記中断は、

中心に前記所望の地点を有し、半径が該論理部の誤差(acc)の値の半分に等しい承認円(circle of acceptance)内に該発射物が位置し、仰角方向及び飛行時間の現

在の値が解として決定されるときに行われるか、または
計算された発射物の位置が所定の境界条件の外側にあるときに行われ、
その後、2つの解が見つかるまで第2の仰角方向を設定することを特徴とする方法

【請求項2】

前記発射速度 (V_{launch}) の少なくとも4倍の値で除算された前記最大誤差 (a_{cc}) として、前記計算部で用いられる時間ステップ (t_{tick}) を最初に計算することを特徴とする請求項1に記載の方法。

【請求項3】

確実に解の最小仰角以下である角度を第1の仰角として固定、例えば、 -90° に固定することを特徴とする請求項1または2に記載の方法。

【請求項4】

軌道上における位置を以下のように反復することを特徴とする方法であって、
 $V_x = V * \text{COS}(\text{deg2rad}) - t_{\text{tick}} * (k_f * V^2 * \text{COS}(\text{deg2rad}) / m)$ 、

$V_z = V * \text{SIN}(\text{deg2rad}) - t_{\text{tick}} * (g + k_f * V^2 * \text{SIN}(\text{deg2rad}) / m)$ 、を、

$X_v = X_v + V_x * t_{\text{tick}}$ 、

$Z_v = Z_v + V_z * t_{\text{tick}}$ 、

$t = t + t_{\text{tick}}$ 、に与え、ここで、

X_v は最も直近に計算されたX方向の位置であり、 Z_v は最も直近に計算されたZ方向の位置であり、

V_x は最も直近に計算されたX方向の速度であり、 V_z は最も直近に計算されたZ方向の速度であり、

【数1】

$$V = \sqrt{V_x^2 + V_z^2}$$

は最も直近に計算され、結果的に得られたXZ平面での速度であり、

$= \text{ATAN}(V_z / (V_x + 1 * 10^{-20})) * \text{rad2deg}$ であり、

ここで、 deg2rad は角度からラジアンへの変換を意味し、 rad2deg はその逆を意味し、

$k_f = C_d * \text{area} / 2$ は、結果的に得られた空気抵抗係数であり (C_d は空気密度に等しい)、

m は質量であり、 g は重力加速度であり、

V_{launch} は V_{launch} に固定され、 V は開始時間 $t = 0$ で V_{launch} に固定されることを特徴とする請求項1～3のいずれか1項に記載の方法。

【請求項5】

前記最も直近に計算されたX方向の位置 (X_v) がX方向におけるターゲットまでの距離 (X_p) よりも大きくなり、前記開始位置とX方向のターゲット位置との距離が0でなくなるまで前記反復処理が続き、その後、第1の解がある軌道についての仰角及び飛行時間で見つかったことが確定することを意味する、前記軌道が前記承認円内にあるか否かの判定、または該軌道が該ターゲットの上にあるかまたは下にあるかの判定がなされることを特徴とする請求項4に記載の方法。

【請求項6】

前記軌道がターゲットの下にある場合、より大きな仰角を新たに選択することを特徴とする請求項5に記載の方法。

10

20

30

40

50

【請求項 7】

前記軌道がターゲットの上にある場合、軌道をターゲットの下にする直前の仰角に戻り、前のステップの増分の一部（例えば、10分の1）である仰角方向のステップの増分でもって軌道に沿った位置及び時間の一連の計算を新たに始めることを特徴とする請求項5に記載の方法。

【請求項 8】

前記解が第1の解である場合、前記第1の仰角が90°、すなわち、真上方向の場合であり、同じ仰角が選択されるときを除いて、別の仰角を選択することによって始まる第2の解の計算を開始することを特徴とする請求項5に記載の方法。

【請求項 9】

前記最も直近に計算されたZ方向の位置（ z_v ）がZ方向におけるターゲットまでの距離（ z_p ）よりも小さくなり、 z_v が0よりも小さくなると共に、前記開始位置とX方向のターゲット位置との距離が0でなくなるまで前記反復処理が続き、その後、第2の解がある軌道についての仰角及び飛行時間で見つかったことが確定したことを意味する、前記軌道が前記承認円内にあるか否かの判定、または該軌道が、X方向において、開始位置から見て、該ターゲットの位置の手前側または向こう側にあるかどうかの判定がなされることを特徴とする請求項8に記載の方法。

【請求項 10】

前記軌道がX方向においてターゲットの向こう側にある場合、より大きな仰角を新たに選択することを特徴とする請求項9に記載の方法。

【請求項 11】

前記軌道がX方向においてターゲットの手前側にある場合、軌道をターゲットの向こう側にした直前の仰角に戻り、前のステップの増分の一部（例えば、10分の1）である仰角方向のステップの増分でもって軌道に沿った位置及び時間の一連の計算を新たに始めることを特徴とする請求項9に記載の方法。

【請求項 12】

前記仰角の増加の選択は仰角の増加とともに減少することを特徴とする請求項6または10に記載の方法。

【請求項 13】

前記計算において、温度、気圧、及び空気湿度に依存して変化する空気抵抗係数（ C_d ）を用いることを特徴とする請求項1～12のいずれか1項に記載の方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、計算された仰角及び飛行時間を使って、軌道上の発射物を距離及び高さが既知の地点において準実時間で作用させる方法に関する。本方法は、発射物を放つための総合システムにおけるPCベースのサポートまたは構成要素のいずれかとして用いることができる。

【背景技術】

【0002】

本明細書において、横方向の位置決め又はアラインメント（方位角）に関しては論じないが、従来技術による方法において、例えば、ターゲットに対する方向を直接測定することによって行われると仮定する。

【0003】

ここで、最適化する方法は、本質的に、軌道に沿った位置及び関連付けられた時点を離散的に適時に計算（discretely timed calculate）する計算部、及び第1の仰角方向を設定し、計算部での計算を監視し、計算された位置が所定の限界値の外側にあるときにその計算を中断し、その後、第2の仰角方向の設定等を行う論理部の2つの部分から構成される。論理部は、仰角方向及び飛行時間の形態で2つの解を決定及び確定する。

【0004】

10

20

30

40

50

この最適化する方法は、擲弾（てきだん）類 / 発射物の空気抵抗力パラメータの特性を特定できる程度に発射試験を受けた軌道システムに用いるためのものである。この方法はまた、空気抵抗力パラメータを実際に特定するためにも用いることができる。初速がより速い発射物については、発射試験によって、温度、気圧、及び空気湿度に対する空気抵抗力の想定し得る依存度を特定することができる。この種の確定された関係に基づいて、このような変化し得る空気抵抗力を本発明の変形例での計算に用いることができるが、これは、各時間ステップにおいてその時の高さが利用できるために可能となっている。

【 0 0 0 5 】

この方法は、ターゲットの測定位置に基づいて、ターゲットに到達させるために、発射装置がどのように向けられるべきかということに対して、迅速かつ選択された精度で応答するために用いることができる。この方法はまた、発砲から擲弾類 / 発射物がターゲットに到達するまでの軌道において必要とされる所要飛行時間についての出力データを供給する。

10

【 0 0 0 6 】

対抗的攻撃手段を用いて敵のミサイルを逸らせたいときには、所望の特定の軌道パターンを生じる戦術的アイデアをもつことになる。このパターンを達成するためには、発射装置をどのように向けるべきか、そして、意図する対抗手段の効果が発揮されるまでの時間を知る必要がある。この戦術的アイデアに基づいて、ターゲット位置を距離、高さ、及び発射方位で示すことは容易であるが、以前より公知の方法を用いてその位置に到達させることは容易ではない。このような対抗システムでは、脅威を発見してから自身の位置、例えば、船等の周囲の所定のターゲット位置で効果が求められるまでの時間は短く、多くの場合において非常に短い。これには、発射装置の位置決め又はアラインメントを計算し、かつ擲弾類の起爆時間を設定するためのシステムが非常に迅速である必要がある。そのようなシステムが本発明の着想の動機であった。しかしながら、本発明はまた、擲弾発射装置及び榴弾砲等の軌道を与える他のシステムにも用いることができるほか、自動火器等を用いて移動するターゲットと戦うための予測アルゴリズムのサポートとして用いることができる。出願人は、本発明たる、この創意ある方法は、全ての用途に関連するという明確な見解を有する。

20

【 0 0 0 7 】

本発明は、距離及び高さ、発射装置を直接制御することが可能な仰角とを置き換えることが可能であることを具体的に示す。擲弾類を変更可能な起爆時間設定で用いることにより、所望の時点において正確な位置に到達させることができる。艦砲の発射装置（naval launchers）を含む例では、チャフ（chaff、レーダ妨害片）を散乱させることができるほか、信号弾の炸薬（pyrotechnic charge）を起こすことができる。

30

【 発明の開示 】

【 発明が解決しようとする課題 】

【 0 0 0 8 】

本発明は、多くの場合において不正確である信頼できない射程図（firing diagrams）の使用に代え、軌道上の発射物を距離及び高さが既知の地点で所望の時点で準実時間で作用させるという課題を解決するものである。

40

【 課題を解決するための手段 】

【 0 0 0 9 】

これは、発明を独立請求項から明らかとなるように構成することにより実現される。本発明の適切な実施形態は残りの請求項より明らかとなる。

【 発明を実施するための最良の形態 】

【 0 0 1 0 】

次に、本発明を添付の図面を参照しながらさらに詳細に説明する。

【 0 0 1 1 】

本発明は、本質的に、計算部及び論理部の2つの部分から構成される（図1参照）。これらの部分は、相互に密接に関連付けられ、結び付けられて一まとめにされているが、こ

50

これらの特性はそれぞれ、ある程度個別に説明することができる。

【0012】

これら2つの部分が正しく起動し連続的に動作することができるためには、これらは最初に8つの初期パラメータ、すなわち、

(種目)	(変数名)	
発射物直径 d	d [m]	
質量	m [K g]	
発射速度	$V_{l a u n c h}$ [m / s]	
空気抵抗係数	C_d	
所望高さの下限値 (考え得るターゲット高さの下限値)	l h [m]	10
出力データの最大誤差	a c c [m]	
ターゲットまでの水平距離	x_p [m]	
ターゲットに対する相対高さ	z_p [m]	

を収集する必要がある。

【0013】

まず、動的フェーズで用いられる時間ステップ $t_{t i c k}$ が計算される。この時間ステップは、論理部での最大誤差 a c c の使用に適合するような大きさに設計される。よって、発射速度 $V_{l a u n c h}$ と最大誤差 a c c との間でどの組み合わせが選択されるかに関わらず、論理部は常に、a c c の大きさに基づいて比較がなされる正確な動作範囲で動作することができる。

20

【0014】

計算部は、常時、ある仰角の軌道に沿った発射物の次の位置を計算する。論理部は、計算部を制御し、例えば、不要な計算を行うことを防止する。論理部は、よって、ある仰角で成功が得られなかった場合には、計算部の計算を中断し、そのかわり、選択された新たな仰角で一連の計算を新たに開始する。また、いくつかの異なる選択可能な方法で、新たな仰角がインクリメント(変数の)値を増加させること)されるように制御する。計算部と論理部の関係は、図2において基本部分が要約されている。

【0015】

図3を参照して、完全な論理チャートを以下に示すが、本発明は、図面上では状態と称する12の異なる条件を用いて説明される。以下の個々の段落では、説明文と共にプログラムコードを示す。

30

【0016】

[状態1]

$X_v = 0.0$ (第1の軌道を確認する前に水平距離の零点規正をする [m])

$z_v = 0.0$ (第1の軌道を確認する前にターゲットに対する高さの初期値の零点規正をする [m])

$t_{t i c} = a c c / (4 * V_{l a u n c h})$ (軌道の離散的(連続的でない)な計算のための時間ステップ [s])

$d e g 2 r a d = / 180$ (換算係数(角度からラジアン))

$r a d 2 d e g = 180 /$ (換算係数(ラジアンから角度))

$p = 1.2$ (空気密度 [g / m³])

$g = 9.81$ (重力加速度 [m / s²])

$a r e a = * d^2 / 4$ (発射物の断面積 [m²])

$k f = C_d * p * a r e a / 2$ (得られる空気抵抗係数)

$f i n d s e c s o l = 0$ (0 : 第1の解を見つける、1 : 第2の解を見つける)

$p a s s f i r s t h i t = 0$ (第2の解の誤検出を防ぐためのフラグ(1 : 関数が起動される))

$n i n e t y d e g r e e s d e t e c t e d = 0$ (90°での検出がなされたときを示すフラグ(初期値の零点規正をする))

40

50

$\theta_1 = 0.0$ (第1の解の仰角(初期値の零点規正をする) [°])
 $time\ of\ flight_1 = 0.0$ (第1の解の飛行時間(初期値の零点規正をする) [s])
 $\theta_2 = 0.0$ (第2の解の仰角(初期値の零点規正をする) [°])
 $time\ of\ flight_2 = 0.0$ (第2の解の飛行時間(初期値の零点規正をする) (s))
 $level\ flag_{30} = 0$ (状態7参照)
 $level\ flag_{60} = 0$ (状態7参照)
 $level\ flag_{70} = 0$ (状態7参照)
 $level\ flag_{89} = 0$ (状態7参照)

10

【0017】

[状態2]

この状態は、確実に第1の軌道を正しく開始させる。

$t_{tick} = 1$ (仰角に対するステップ変数の初期設定)
 $launch = -90$ (仰角 $launch$ の初期値)
 $state = 3$ (次の状態 = 状態3)

【0018】

[状態3]

$launch$ を新たに調整するごとに、以下のステップを行うことが必要となる。この状態は、状態2、7又は11のうちの1つから起動される。

20

$t = 0.0$ (新たな軌道の各々の前に時間の零点規正をする。)
 $X_v = 0.0$ (次の軌道の前に水平距離変数の零点規正をする。)
 $Z_v = 0.0$ (次の軌道の前に(ターゲットに対する)高さ変数の零点規正をする。)
 $state = 4$ (次の状態 = 4)

【0019】

[状態4]

この状態は、状態3、5又は12のうちの1つから起動される。時間 $t = 0.0$ において、 V は、その時の軌道についての初期値が与えられる。

【0020】

$if(t == 0.0)$
 {
 $= launch$ (姿勢変数)の初期値は $launch$ を基にして与えられる。
 $V = V_{launch}$ (V (軌道速度変数)の初期値は V_{launch} を基にして与えられる。)
 }

30

【0021】

次いで、現在の軌道上の次の位置が計算される。

$V_x = V * \cos(\theta * deg2rad) - t_{tick} * (k_f * V^2 * \cos(\theta * deg2rad) / m)$
 $V_z = V * \sin(\theta * deg2rad) - t_{tick} * (g + k_f * V^2 * \sin(\theta * deg2rad) / m)$

40

【数2】

$$V = \sqrt{V_x^2 + V_z^2}$$

50

```

    = ATAN ( Vz / ( Vx + 1 * 10-20 ) ) * rad2deg
Xv = Xv + Vx * ttick
Zv = Zv + Vz * ttick
t = t + ttick
(ここで、deg2radは、角度からラジアンへの変換を意味し、rad2degは
その逆を意味する)
【0022】
if ( ninetydegreesdetected == 0 )
{
    if ( zv > zp ) ( 90°の仰角がまだ検出されておらず、かつzvが
zpよりも大きくなった場合、テストを行って投射 ( throw ) が真上に行われるか否かを
確認する。 )
    {
        if ( ( xp == 0 ) && ( launch > 89.9 ) )
        {
            t1 = 90.0 ( 90°の仰角についての第1の解
が見つかる。現在の時間が収集される。 )
            timeofflight1 = t
            ninetydegreesdetected = 1
( 仰角90°が検出される )
            state = 12
        }
    }
}
else
{
    if ( zv < zp ) ( 前回のループにおいて90°が検出された場合、発
射物がzpの真上に位置する。zv < zpであるとき、完全に確実にするためにさらなる
チェックがなされる。 )
    {
        if ( ( xp == 0 ) && ( launch > 89.9 ) )
        {
            t2 = 90.0 ( 90°の仰角についての第2の解
が見つかる。現在の時間が収集される。 )
            timeofflight2 = t
            state = 12
        }
    }
}
if ( zv < ( lh - ( 2 * acc ) ) ) ( 発射物が高さの下限値を越えるか否か及び
その時期を決定するためのチェック。「 - ( 2 * acc ) 」は zp = lhの場合に用いら
れる。 )
    state = 7 ( 高さの下限値を越える。次の状態 = 状態7 )
}
else
{
    state = 5 ( 次の状態 = 状態5 )
}
【0023】
[ 状態5 ]

```

10

20

30

40

50

この状態では、仰角 90° を有さない解を探す。

if (findsecsol == 0) (第1または第2の解を探しているかに応じて選択。第1の解を探す。)

```

{
    if ( ( xv > xp ) && ( xp <> 0.0 ) )
    {
        state = 6 ( 発射物は xp を越える。90°ではない。次の状態 = 状態6 )
    }
    else
    {
        state = 4 ( 全体的な判定のためのデータではない。軌道の確認を継続する。次の状態 = 状態4 )
    }
}
else
{
    if ( ( < 0.0 ) && ( Zv < Zp ) && ( Xp < 0.0 ) ) ( 第2の解を探す。発射物は、90°ではない負の姿勢を有し、かつ zp を越えているか? )
    {
        if ( Xv < Xp ) ( 下降移動中に Zp を越し、かつ同時に、それが xp の前に起きたか? )
        {
            state = 9 ( Yes の場合、発射物が探していた位置 ( xp, zp ) に対してどれだけ近接しているかを調べる。次の状態 = 状態9 )
        }
        else
        {
            state = 7 ( No の場合、下降移動中に xp の先で zp を越える。次の状態 = 状態7 )
        }
    }
    else
    {
        state = 4 ( 全体的な判定のためのデータではない。軌道の確認を継続する。次の状態 = 状態4 )
    }
}
}
【 0 0 2 4 】
[ 状態6 ]
この状態は、状態5からのみ起動され得る。
if ( zv > zp ) ( 発射物は、xp を越え、それと同時に zp の上を越えたか? )
{
    state = 9 ( Yes の場合、発射物が探していた位置 ( xp, zp ) に対してどれだけ近接しているかを調べる。次の状態 = 状態9 )
}
else
{
    state = 7 ( 軌道は、第1の解を探し求めてサーチした位置をはるかに下回っている。次の状態 = 状態7 )
}
}

```

10

20

30

40

50

【0025】

[状態7]

$l a u n c h$ の各値が解に辿り着かない場合は、結果として、この状態が起動される。この状態は、適切な軌道を再度新たに実行することができるように $l a u n c h$ をインクリメントする。 $l a u n c h$ の値の大きさに応じて、適切にインクリメントされる。 $t i c k$ の値が大きすぎる場合、最終的な解は得られない。発射物の経路は、この状態の論理（ロジック）では単純に決定的な段階に達しない。値が低すぎる場合、タスクを解決するために必要な時間の消費が急激に増加する。 $l a u n c h$ が大きくなればなるほど、エラーが発生する危険性を完全になくすことができるように、 $t i c k$ は低くなる必要がある。

10

【0026】

```

I f (   l a u n c h > 3 0 )
           L e v e l f l a g 3 0 = 1
I f (   l a u n c h > 6 0 )
           L e v e l f l a g 6 0 = 1
i f (   l a u n c h > 7 0
           l e v e I f l a g 7 0 = 1
i f (   l a u n c h > 8 9 )
           l e v e l f l a g 8 9 = 1
           t i c k = 1 0 - l e v e l f l a g 3 0 * 7 - l e v e l f l a g 6 0 * 2 - l e
v e l f l a g 7 0 * 0 . 6 - l e v e l f l a g 8 9 * 0 . 3
           l a u n c h = l a u n c h + t i c k

```

20

【0027】

```

i f (   l a u n c h > 9 0 . 0
{
           s t a t e = 8 ( 探した位置 ( xp , zp ) は、投射範囲 ( throwing ran
ge ) の外側に位置する。次の状態 = 状態 8 )
}
e l s e
{
           s t a t e = 3 ( 新たな軌道の実行。次の状態 = 状態 3 )
}

```

30

【0028】

[状態8]

探した位置 (x_p, z_p) は、投射範囲の外側にある。飛行角度および時間は、適切に 0.0 の値が与えられる。この状態が起動されると、状態の全処理が終了し、以下の結果が得られる。

```

_1 = 0 . 0
t i m e o f f l i g h t _1 = 0 . 0
_2 = 0 . 0
t i m e o f f l i g h t _2 = 0 . 0

```

40

【0029】

[状態9]

この状態は、解を見つけるために逐次近似を始める必要があると判定されたとき（5参照）、または第2の解が誤ることを防ぐ必要があるときのいずれかに起動される。ここでは、解が見つかったときも判定される（4参照）。

【0030】

まず、見つかった位置と現在の位置との半径方向の誤差が計算される（下記の1参照）。状態12では、第1の解が見つかったときに「 $p a s s f i r s t h i t$ 」フラグが1に設定される。軌道における次の位置を計算した直後に、状態9が起動され、この場合

50

には、「diff」が「acc/2」よりも小さい可能性が高い。誤った第2の解が間違っ
て検出されることを防ぐために、この状態を中断し、そのかわりに、状態7に進む(3
・参照)。

【0031】

最終的に、最も第2の解である可能性があるものが承認可能であるか評価される
とき、2.では、「passfirsthit」が現在まで構成してきた停止状態が解放される
ことを確認する。

【0032】

【数3】

$$diff = \sqrt{(x_v - x_p)^2 + (z_v - z_p)^2} \quad 1. \quad 10$$

```

if (diff > (acc / 2) && passfirsthit == 1)      2.
    passfirsthit = 0
if (diff > (acc / 2) && passfirsthit == 1)      3.
{
    state = 7
}
else                                           20
{
    if (diff < (acc / 2))
    {
        state = 10      4.
    }
    else
    {
        state = 11      5.
    }
}
}                                           30

```

【0033】

[状態10]

この状態は、状態9からのみ起動され得る。そして、90°ではない解が見つかる。「
findsecsol」=0である場合(すなわち、第1の解が見つかる前)、 t_1 及び
timeofflight $_1$ には、それぞれ、 $launch_1$ 及び t の瞬時値が与えられ
る。「findsecsol」=1の場合には、 $launch_2$ 及びtimeofflight $_2$ に
は対応する値が与えられる。

【0034】

図3のフローチャートから、「findsecsol」=1のとき、状態10により、
その解の値が解2に直接与えられ、全ての実行が終了することが明らかである。同時に、
下記のコードから、第1または第2の解のいずれが送られたかに関わらず、状態10は常
に状態12に直接進むことが明らかである。この場合、いずれにしても、この相違は重要
ではない。各状態1~12について提示されたコードラインは、実際、C++で記述され
たアプリケーションから直接抽出されたものである。また、プログラムは関数的な方法で
終了できることが必要であると同時に、フローチャートは、関数を十分明確に記述する
ことが可能なものでなければならない。

【0035】

```

if (findsecsol == 0)
{

```

```

    launch = launch

```

50

```

        timeofflight1 = t
    }
    else
    {
        2 = launch
        timeofflight2 = t
    }
    state = 12
【0036】
    [状態11]
    この状態は、状態9からのみ起動され得る。
【0037】
    状態9は、仰角が探した位置( $x_p, z_p$ )を通過する直前に確立される。それゆえ、この探索は、まず、1ステップ逆行する必要がある(下記の1.参照)。次いで、 $t_{ick}$ は、10分の1だけ縮小される(2.参照)。このようにして、元の増分の1/10だけが実行される(3.参照)。次の軌道の仰角について、仰角が点( $x_p, z_p$ )の上にあるのか、または下にあるのかに応じて、状態7の通常の $t_{ick}$ と、ここで行われる縮小とは交互に協調しあう。このようにして、正しい解を決して逃さない一種の逐次近似が常に提供される。
【0038】
    launch = launch - tick
    tick = tick / 10
    launch = launch + tick
    state = 3
【0039】
    [状態12]
    この状態に入ったときにfindsecsolがまだ0である場合、第1の解のみが見つかっている。findsecsol及びpassfirshitはまず1に設定される。次いで、90°が検出されたか否かが確認される。検出されている場合、処理は状態4に進み、垂直方向における軌道の次の位置が計算され得る。
【0040】
    ninetydegreesdetected = 0である場合、処理は状態7に進み、次の仰角の確認が開始され得る。状態12に入ったときにfindsecsol = 1である場合、処理全体が終了する。ターゲットの位置及び特性パラメータについて利用可能な全ての想定可能な解は、この段階では、状態4、8、又10においてすでに解かれている。
【0041】
    if (findsecsol == 1)
        break
    findsecsol = 1
    passfirshit = 1
    if (ninetydegreesdetected == 1)
    {
        state = 4
    }
    else
    {
        state = 7
    }
【0042】

```

10

20

30

40

50

図3を参照して本発明の一実施形態について説明したが、以下に、 xz 平面上の軌道の2つの位置に発射物を示す図4を参照して説明及び所見を述べる。これらの発射物の位置での加速度及びその速度を示す。

【0043】

第1の位置計算の前に、 $(\theta = \theta_{\text{launch}})$ 及び V ($V = V_{\text{launch}}$)に初期値が与えられる。 V_x 及び V_z の計算では(状態4参照)、 θ 及び V の前の値を用いて近似が行われる。次いで、 V_x 及び V_z について、 θ 及び V の新たな値が計算される。そして、 X_v 及び Z_v の単純な更新が行われる。最後に、 t が上方方向に調整される。

【0044】

図4における発射物の加速度 a は、

【数4】

$$a = \frac{f}{m}$$

と表すことができ、この場合の f は、空気抵抗

【数5】

$$f = -k_f * V^2$$

によって生じる反作用力である。よって、反対に作用する加速度は、

【数6】

$$a = -\frac{k_f * V^2}{m}$$

と表すことができ、これは、水平方向の加速度成分 $a_x = -k_f * V^2 * \cos(\theta * \text{deg}2\text{rad}) / m$ 及び垂直方向成分 $a_z = -k_f * V^2 * \sin(\theta * \text{deg}2\text{rad}) / m$ を与える。

【0045】

最初に、時間ステップ t_{tick} が計算され、 a_{cc} 及び V_{launch} について最適化される。 $t_{\text{tick}} = a_{cc} / (4 * V_{\text{launch}})$ となるように t_{tick} の大きさを決定することによって、2つの隣接する位置間の半径距離は a_{cc} よりも大きくなる。それゆえ、 a_{cc} によって、2つの解の各々についての最終的な結果における最大誤差を十分に決定することができる。これには、空気抵抗力の作用について、非常に短い時間ステップを用いて軌道にある物体の古典的な微分方程式と比較する場合、この離散的な計算方法自体が十分に正確である必要がある。

【0046】

t_{tick} の計算において分母が2ではなく4を含むのは、仰角及び飛行時間の解がかなり正確であることを保証するために処理されなければならない2つの異なる誤差原因が存在するためである。1つは、古典的な微分方程式と本明細書中に記載の離散的な方法との計算の誤差によって生じ、誤差は $a_{cc} / 2$ よりも大きくはならない(次の段落を参照)。時間 t_{tick} 中の軌道における飛行経路を、 a_{cc} の $1/2$ ではなく、最大で $1/$

10

20

30

40

50

4にすることができる t_{tick} を用いることによって、最大計算誤差を $acc/2$ に低減することができる。

【0047】

第2の誤差原因の保証された最大誤差は $acc/2$ であり、この値に対して状態9における全ての比較がなされる。これによって、各解が仰角及び飛行時間によって確認されるとき、半径 = acc であり、かつその中心が入力データ、すなわち、 (x_p, z_p) として示された位置に正確に配置される虚円内で軌道は確実に終了する。

【0048】

本発明は、高さによって変化する風力、風向き、空気密度等の異なる付加的要因を、種々の方法で考慮することによって発展させることができる。基本的に、これらの場合においても図3のフローチャートが用いられる。このとき、軽微ながら変更が必要とされる。

10

【0049】

本発明の精度を確認するために、本明細書中において提示された基本形式では、タスクについて作成された2つの方法によって考察した。第1の方法は、初期の、離散的かつ派生的なブロック (initial, discrete and derivative blocks) に、所望の目的のために個々のプログラムコードが与えられた、時間連続関数をシミュレートすることが可能なプログラム ACSL (高度連続型シミュレーション言語 (Advanced Continuous Simulating Language)) で作られたシミュレーションモデルである。第2の方法は、Visual C++ 6.0、MFC Wizard でプログラムされた発明を含む。

【0050】

20

非常に多くのシミュレーション及び実行が行われた。そして、これら2つの方法および Mathcad 2000 というプログラムで確認された軌道の古典的な微分方程式から得られた結果が比較された。各々の比較では、最終的な位置の全てが、中心位置 (x_p, z_p) を有する半径 acc 円内にあった。

【図面の簡単な説明】

【0051】

【図1】本発明を計算部及び論理部へと基本的に区分する様子を示す図である。

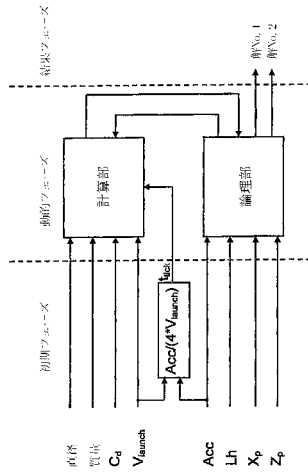
【図2】図1の計算部及び論理部の基礎レベルの構成を示す図である。

【図3】本発明の完全なフローチャートを示す図である。

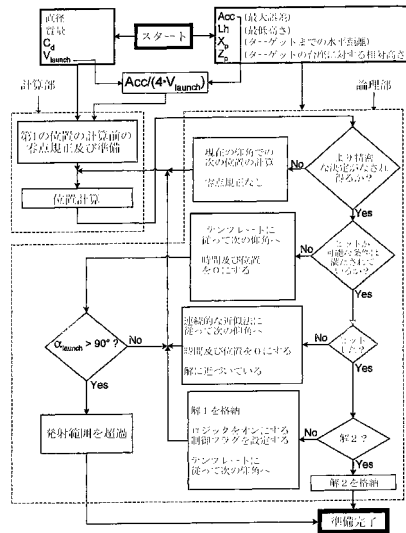
【図4】 xz 平面の軌道上における発射物、並びに2つの近接した時点における発射物の加速度、速度及び関連付けられたベクトルを示す図である。

30

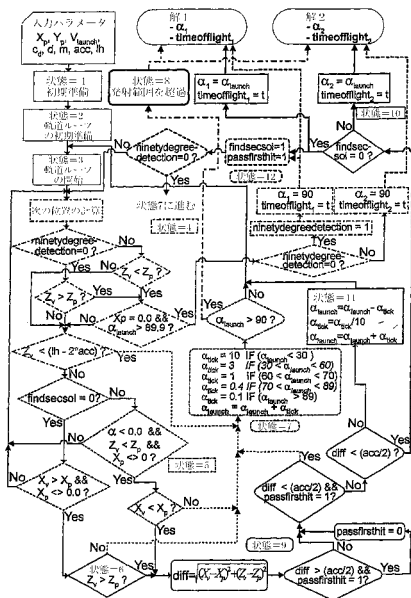
【図1】



【図2】



【図3】



【図4】

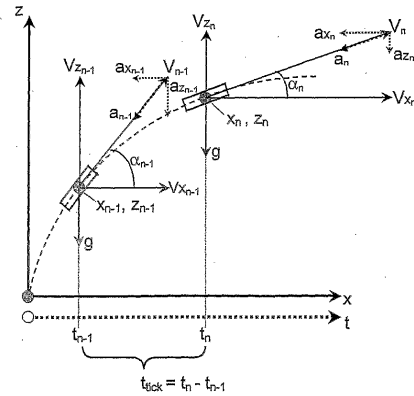


Fig. 4

フロントページの続き

(74)代理人 100125586

弁理士 大角 菜穂子

(72)発明者 ストランド, パトリック

スウェーデン王国、エス 5 9 5 9 6 ミエルビー、シャ、ハイディンイエヴェーゲン 1 1

審査官 見目 省二

(56)参考文献 特開2000-249496(JP, A)

特開平11-166799(JP, A)

特開平02-267497(JP, A)

(58)調査した分野(Int.Cl., DB名)

F41G 3/00