

分割主鏡シミュレーター — パイロット版設計 II

岡山新技術望遠鏡グループ

平成 22 年 5 月 26 日

目次

1	概要	1
2	仕様	1
2.1	設定項目	1
2.2	入力データ	2
2.3	出力データ	2
3	設計	2
3.1	定数定義	2
3.2	入出力	3
3.2.1	入力共有メモリデータ構造	3
3.2.2	設定共有メモリデータ構造	3
3.2.3	出力共有メモリデータ構造	3
3.3	動作	4

1 概要

分割主鏡の制御システムをどう構築するかを検討・検証のため、モデルシミュレーションが可能になるようなシミュレータを構築することを目的とする。

このパイロット版では、セグメントの動きを把握するという目的の元、与えられたアクチュエーターの位置に対する分割主鏡のパラメータ、および指定された位置に設置したギャップセンサーの値を出力するモジュールを作成する。

2 仕様

2.1 設定項目

設定項目は以下のものがあげられる。

セグメントごとのギャップセンサー取り付け位置 ギャップセンサーを取り付ける位置をセグメントごとに指定できなければならない。最終的に全セグメントに対して同じ位置に取り付ける可能性はあるが、さまざまな仮想実験を行うために、全セグメントに対して任意個数の任意位置に設定できるようにしておくのが望ましい。

ギャップセンサーペアの対応関係 上の設定項目で設定されたギャップセンサーの配置に対して、一つのギャップセンサーを構成するような二つの取り付け位置を設定できるようにする。

片側固定のギャップセンサー ギャップセンサーの2固定位置のうち、片側がセグメントでなく主鏡セルに固定された z 方向に移動しない固定点であるようなセンサー位置を実現できる必要がある。

なお、本シミュレータでは上記二つの設定項目がコンシステントであることを確認することについては、ソフトウェアの仕様としては含まない。

ただし、上記で任意個数としている設定数について、ソフトウェアの設計を簡単にするために、それぞれ、取り付け位置については一つのセグメントに対して設定できる個数、ギャップセンサーペアの対応関係についてはギャップセンサー個数、のそれぞれについての上限を設ける。

2.2 入力データ

アクチュエーターの理想的原点に対する相対現在位置を本シミュレータの入力データとする。この理想的原点は、すべてのアクチュエーターが原点にある場合に鏡面が理想形状を実現するような点をいう。実際の望遠鏡においてはSH/PCS系を利用して光学的にキャリブレーションを行った際に得られる点となる。

2.3 出力データ

出力は、セグメントの配置について計算できるすべてのパラメータを外部から参照できるようにする。これに当てはまるものを書き上げると以下ようになる。

背面に対する法線単位ベクトル

鏡面側の中心点における法線単位ベクトルに一致し、SH カメラで光学的に試験した場合に得られるデータに相当する。

全ギャップセンサー取り付け位置の z 値

指定した取り付け位置について z 座標の値はデバッグ・チェックのために必要である。

全ギャップセンサーの出力値

シミュレータの目的となる値である。

なお、将来的に傾けた場合における機械変形などを考慮したシミュレーション機構にもできるように、固定位置とギャップセンサーを分けておく。こうしておくことで、ギャップセンサーの読み出し値計算部分を分離して、アクチュエーター固定点やギャップセンサー固定位置における機械変形を入れた計算や、ギャップセンサーそのものの環境(気温・湿度など)による影響を導入したりする、などが容易にできるようになる。

3 設計

3.1 定数定義

望遠鏡の設定や、動作設定のためにいくつかの定数を定義している。これらの定数はすべて `common.hh` に記述されている。

A: `divm1_sim_segments` 全セグメント数。通常 18 である。セグメント ID は $0 \sim A - 1$ の値で表される。

B: `divm1_sim_ipc_pos_max` 定義可能な全センサー固定位置の数。各セグメントに対しての固定位置であるので、ギャップセンサー一つにつき二つの固定位置が必要であることに注意。なお、固定位置を参照するパラメータについてのみ、0 は完全固定された固定点を参照する特殊位置、 $1 \sim B$ が固定位置の ID であるとする。(実際は別途変数として定義される有効な B の値 (`valid-B`) まで。)

C: `divm1_sim_ipc_gap_max` 定義可能な全ギャップセンサーの数。センサー固定位置二つをペアにして一つのギャップセンサーを構成する。その組み合わせを定義する。ギャップセンサー ID は $0 \sim C - 1$ の値で表される。

3.2 入出力

入出力はすべて IPC/SHM (共有メモリ) から読み出し、もしくはへ書き込みを行うことによる。
本パイロット版では、共有メモリを入力・出力・設定の3種類持つ。

3.2.1 入力共有メモリデータ構造

入力共有メモリは SHM_Actuator クラスが取り扱う。データは各セグメントごとに3個あるアクチュエーターの制御位置である。

すべて値は自由に取れ、配列のインデックスは $0 \sim A - 1$ である。

double *head_value_p1 セグメント内の1番目のアクチュエーターの制御位置。

double *head_value_p2 セグメント内の2番目のアクチュエーターの制御位置。

double *head_value_p3 セグメント内の3番目のアクチュエーターの制御位置。

3.2.2 設定共有メモリデータ構造

この共有メモリは SHM_SegConf クラスが取り扱う。データはセンサー固定位置とギャップセンサーの定義である。

valid-B: size_t valid_pos_count 定義されているセンサー固定位置の数。

unsigned char *head_seg 定義しているセンサー固定位置がどのセグメント上にあるかの定義。値はセグメント ID である $0 \sim A - 1$ で、配列のインデックスは固定位置 ID から1引いた $0 \sim valid - B - 1$ である。

double *head_value_x センサー固定位置の x 座標。値は自由に取れ、配列のインデックスは固定位置 ID から1引いた $0 \sim valid - B - 1$ である。

double *head_value_y センサー固定位置の y 座標。値は自由に取れ、配列のインデックスは固定位置 ID から1引いた $0 \sim valid - B - 1$ である。

valid-C: size_t valid_gap_count 定義されているギャップセンサーの数。

unsigned short *gap_id_a ギャップセンサーを構成するセンサー固定位置一つ目。値は完全固定点を除いた固定位置 ID の $1 \sim valid - B$ であり、配列のインデックスはギャップセンサー ID である $0 \sim valid - C - 1$ である。

unsigned short *gap_id_b ギャップセンサーを構成するセンサー固定位置二つ目。値は固定位置 ID の $0; 1 \sim valid - B$ であり、配列のインデックスはギャップセンサー ID である $0 \sim valid - C - 1$ である。

3.2.3 出力共有メモリデータ構造

この共有メモリは SHM_Segstat クラスが取り扱う。データは各セグメントに対する鏡面中心での単位垂線ベクトル、センサー固定位置の z 座標値、ギャップセンサーの値である。

double *normal_vector_x 各セグメントに対する鏡面中心での単位垂線ベクトル x 。値は自由に取れ、配列のインデックスはセグメント ID の $0 \sim A - 1$ である。

double *normal_vector_y 各セグメントに対する鏡面中心での単位垂線ベクトル y 。値は自由に取れ、配列のインデックスはセグメント ID の $0 \sim A - 1$ である。

double *normal_vector_z 各セグメントに対する鏡面中心での単位垂線ベクトル z 。値は自由に取れ、配列のインデックスはセグメント ID の $0 \sim A - 1$ である。

`size_t valid_pos_count` 定義されているセンサー固定位置の数。(valid-B)

`double *head_value_z` センサー固定位置の z 値。値は自由に取れ、配列のインデックスは固定位置 ID から 1 引いた $0 \sim \text{valid} - B - 1$ である。

`size_t valid_gap_count` 定義されているギャップセンサーの数。(valid-C)

`double *gap_value` ギャップセンサーの値。値は自由に取れ、配列のインデックスはギャップセンサー ID の $0 \sim \text{valid} - C - 1$ である。

3.3 動作

シミュレーター本体はループを実行し続ける構成とし、あるタイミングでの共有メモリのデータを参照して演算した結果を出力用共有メモリに書き込む。

シミュレーターの補助プログラムとして、共有メモリのデータを参照するステータスマニター、およびアクチュエーター駆動のコマンドを発行するためのコントローラを提供する。