

もっか、繰り返しパターンを発見するアルゴリズムを考えています。繰り返しだからフーリエ変換かなと思ったりしましたが、今は地道にパターンマッチングしながらやっています。

- (1) 点は `MapData` というオブジェクトで管理し、`MapSystemObjectBase` という2次元マップを作って、配置を管理しています。
- (2) 線は `MapData` の連鎖で `MapDataSequence` というオブジェクトにして管理しています。基本的に線は端点や分岐点などの特異点で区切っています。あとは断線も線として管理しています。そんなものが図面全体の線分列として、`MapDataSequenceSet` として管理することとしています。
- (3) そこで、繰り返しパターンですが、この線と特異点の配置が繰り返すという考え方でとらえていこうとしています。まずは、2つの接続する `MapDataSequence` を順次選んで、`MapDataSequenceCluster` としていき、2つの `MapDataSequenceCluster` のマッチングを行って、一致したら、`MapDataSequenceClusterSet` に登録していく。同じパターンの `MapDataSequenceCluster` のものは一つの `MapDataSequenceClusterSet` に登録する。そうしていくと、異なるクラスターセットがいくつかできます。それらを `MapDataSequenceClusterSetList` オブジェクトで管理する。
- (4) そうして、今度は `MapDataSequence` を3つ連続するものを調べます。今、4つ、5つとクラスターを大きくして行きます。それらを `MapDataSequenceClusterSetListSequence` オブジェクトで管理する。そうすると、繰り返しパターンの状況が `MapDataSequenceClusterSetListSequence` の配下に集約できるわけです。

パターンマッチングですが、線分の長さは  $3/4$  くらい一致していればいいかなと。2線分のなす角度は、2線分の交点が2線分の離れた2点を結ぶ線分の右か左かが一致していること・・・でいいかなと思っています。

おわり