門 部

競技部門

No. 1 登録番号

30054

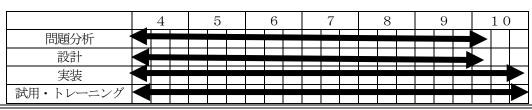
No.2

**フリガナ** ||タタミミズク

チーム名 ||タタミミズク

1) 予定開発期間:6ヶ月 2) 予定開発人数: 4人

No.3



### 実現方法

# 1. はじめに

今年度の競技内容は、様々な形状のピースを敷き詰める問題である。これは2次元パッキング問題(詰 め込み問題)に帰着させて考えることができるが、一般的に多項式時間では解けないと予想されている。 そこで、計算量を落として競技時間内に近似解を導出することを目標とする。具体的には、順位決定方法 に基づき、空きマスができるだけ少なくなるように探索する。

## 評価関数

ピースの敷き詰め方を全通り調べるのは困難なので、ピースの置き方の良し悪しを判断する評価関数を 作り、悪い置き方を切り捨てていく。この時の評価基準はいくつか考えられ、それを書いていく。

### 2.1. 隣接する辺

隣接する辺が多くなるように置くと、結果的に無駄のない配置に置かれる傾向にあることが実験より分 かった。試しに盤面に配置し、そのピースの周辺の辺の数を数える。

## No.4

# 2.2. 大きいピースから使う

盤面は徐々に埋まっていき、大きいピースは使いにくくなっていく。そこで、大きいピースを配置する ことを意識した探索を行う。大きいピースのみ先に探索し、配置する場所が決まったら予約しておく。次 に小さいピースの探索を行い、予約された領域には置かないようにする。

# 2.3. 空白領域の数

空きマス数が同じでも、小さな空白領域(空白領域=連続した空きマス)がたくさんあるのと、大きな 空白領域が1つあるのとでは後者のほうが新しいピースを配置しやすい。つまり、空白領域の数が少なく なるように探索するとよい。試しに盤面に配置し、盤面上の空白領域の数を数える。

### 3. サンプルケース

試用・トレーニングのためにサンプルケースをいくつか作る。ランダムなケースと人為的なケースを用 意し、それぞれに強いプログラムを書く。

### 開発環境

No.5

言語: C++/C#/Python/Ruby/JavaScript/Japanese/English エディタ: Visual Studio, Sakura Editor, Vim, notepad

全国高等専門学校 第26回プログラミングコンテスト:長野大会