

地区: 北海道 地域: 北海道函館市



チームNo. 160 チーム名: 恵庭戦隊トシキング 所属: 公立はこだて未来大学

チーム紹介、目標、意気込み

2015年から恵庭を守るため、悪の組織〜デスマーチ団〜と戦い続けたトシキング、今年に入り活躍していた先輩隊員が去り、チームを設立した初期隊員を知っているメンバーはもういない。このあたりで一度、ボロボロになったモデルベースを見直さないといけない。走りのトシキングから、モデルと走行にバランスをとったトシキングに方向転換を試みる。今年はデスマーチ団の放ったコロナの影響で、新規メンバーも集まりづらく苦戦を強いられている。それでも負けるなトシキング!

モデルの概要

モデル	概要
要求モデル	今年はコロナによる影響も要求図に取り入れ,ブロックビンゴ競技における非機能要求に具体的な視点を取り入れた.
分析モデル	ゲームの構成要素では、ブロックビンゴエリアの分析をした結果、新たな要素を発見したため、クラス図に反映している.
設計モデル	要求モデルで発見した非機能要求を活かし分析モデルの指針をもとに設計の方針に反映した.

モデルの構成

- 1. 要求モデル
- 今のトシキングに足りないものとして、非機能要求の中で見つけた「モデルとプログラミングを紐づけた開発をする」ため、 一から開発することを決め、以下の要求を洗い出した
- 分析ではブロックビンゴエリアの分析は行うが、経路探索はしない
- ・ 走行体の動作定義は過去のモデルでは多く定義しすぎたため最 小限の定義にした
- 設計において、継続して使用できる走行体システムを考えた
- 2. 分析モデル
- モデリングの訓練としてブロックビンゴエリアの分析をし、規 約以外からも新しいクラスを発見することができた
- 走行体の動作定義では、動作を制御するプログラムはあったが モデルがなかったため、解読できなくなり一から作りモデルに も適応する
- 指針・解法では、開発リソースが足りないことを要求モデルで見つけたため、経路探索を開発しないことにした。ただし、走行体の動作のみで攻略できる方法を考えた
- 3. 設計モデル
- 走行体システムを継続して使えるようにシステムへの要求と 走行体への要求で考えた。その結果システムへの要求をEV3 の状態管理・競技の状態管理で実現し、走行体への要求を動 作モデル・判別モデル・デバイスドライバで実現した

1

要求モデル

1.1 目標設定

競技規約より、今年の目標設定を図1.1.1に示す。また、各競技における時間配分を表1.1.1に示す。表1.1.1より走行競技30秒の目標設定理由は、去年のCS大会コースにおいて全長と各チームのタイムから平均速度を算出すると38cm/sだった。今年の大会は2019年のコースより長くコーナー数も多いため平均速度35cm/sで走行することを決めた。



図1.1.1 今年の目標設定

表1.1.1 各競技における時間配分

競技	時間	距離	平均速度
走行競技	30s	1017cm	35cm/s
ブロックビンゴ競技	70s	不明	15cm/s
ガレージ競技	20s	140cm	10cm/s

1.2 機能要求

1.1より目標を達成するために必要な機能を図1.2.1に示す.

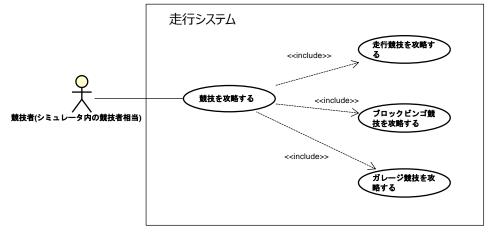


図1.2.1 機能要求発見のためのユースケース図

1.3 非機能要求

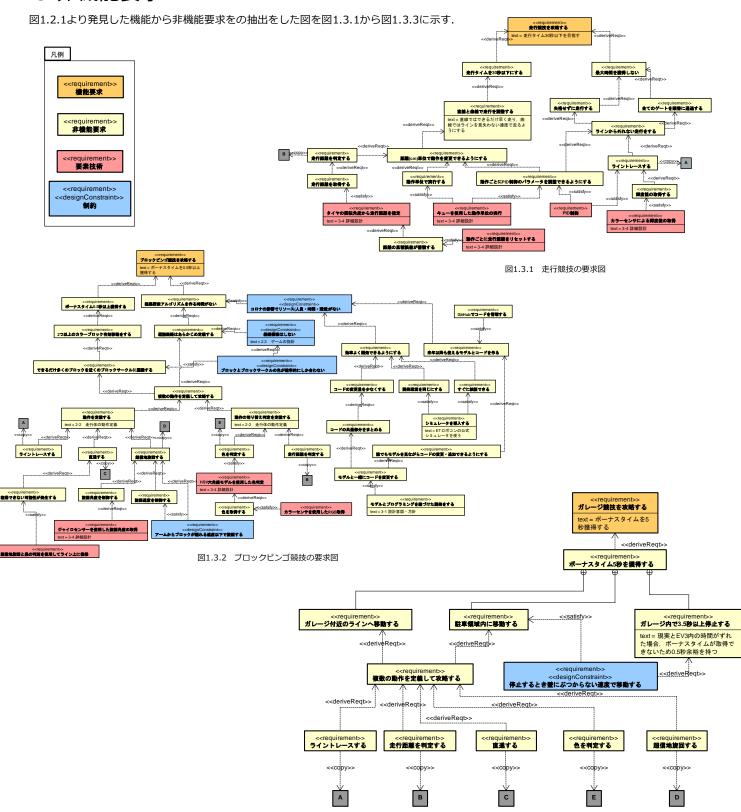


図1.3.3 ガレージ競技の要求図

2 分析モデル

夏庭戰隊

2.1 ゲームの構成要素

まず、競技規約に掲載されているブロックビンゴエリアを図2.1.1に示す. 分析の結果, ビンゴの表現とボーナスサークルの表現をするために, 「ミニブロックビンゴエリア」・「ビンゴ行」・

「ビンゴ列」・「非ボーナスサークル」が必要であると考えた.

ブロックビンゴエリアの構成要素を分析・整理した結果を図2.1.2に示す.

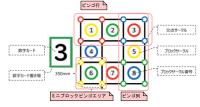


図 2.1.1 ブロックビンゴエリア

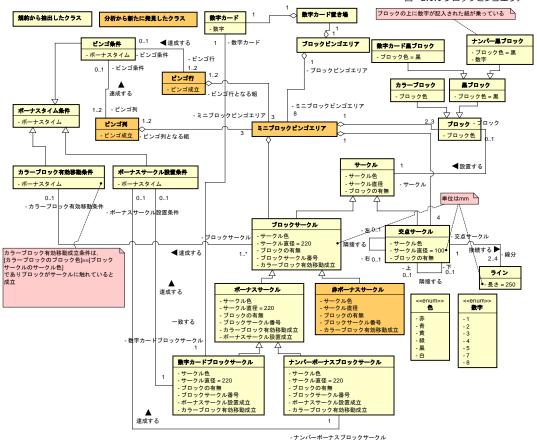


図 2.1.2 ブロックビンゴエリアの構成要素(クラス図)

2.2 走行体の動作定義

競技規約より、ブロックビンゴ競技に必要な動作を図2.2.2、図2.2.3に示す。また、動作を実現させるために 必要な動作モデル, 判定モデルをA~Fで定義し表2.2.1, 表2.2.2に示す.

走行体の定義

走行体を三角形で表し 前方と後方は図2.2.1のように定義する

後 前

図2.2.1 走行体の定義 動作 判定 F B D

表2.2.1 走行体の動作モデル

動作モデル	説明
A. ライントレース	線分上を直進するおよび,復帰すると きに使用する
B. 直進 (前進・後退)	交点サークル上を直進するときに前進 を使用する ブロックサークル内にブロックを搬入 するときに前進・後退を使用する
C. 超信地旋回	その場で旋回するときに使用する 45度刻みに左右180度旋回可能とする

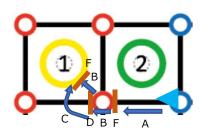


図2.2.3 ブロック運搬動作の例示

表2.2.2 走行体の検知モデル

判定モデル	説明
D. 距離	ライントレース動作または <u>直進動作</u> と 組み合わせて動作の終了条件にする
F. 色	ライントレース動作または <mark>直進動作</mark> と 組み合わせて動作の終了条件にする

2.3 ゲームの指針

図1.3.2より, 交点サークル上にあるブロックを近くのブロックサークルへ運搬し, ボーナスタ イム0.5秒以上を狙う

指針を立てるときに発見した課題と解決策を表2.3.1に示す.

表2.3.1 指針を決めたときに発見した課題と解決策

課題	解決策
ブロックを保持したまま左右に旋回す るときブロックがアームから離れてし まう可能性がある	ブロックがアームから離れない旋回速度を求める
交点サークルを左右に旋回するとき旋 回角度のずれによりラインに復帰でき なくなる可能性がある	ラインの色が黒のため, 黒を発見するまで直進し, ライントレースすることによりライン上に復帰する
交点サークルを直進するときラインに 復帰できなくなる可能性がある	交点サークル直進後,黒を発見できなければ,その場で左右に45度ずつ車 体を超信地旋回することによりラインを発見し復帰する

2.4 ゲームの解放

2.3の指針を達成するために、経路探索を導入する必要がないと判断した、理由は、2.1より交点サークル上に おいてブロックの位置は固定されており、2.2より、走行体の動作も定義されていることにより、運搬経路は あらかじめ定義できると判断したためである. したがって, LRコースにおける運搬経路を図2.4.1に示す. L コースでは時計回りに, Rコースでは反時計周りにブロックを順次運搬する.

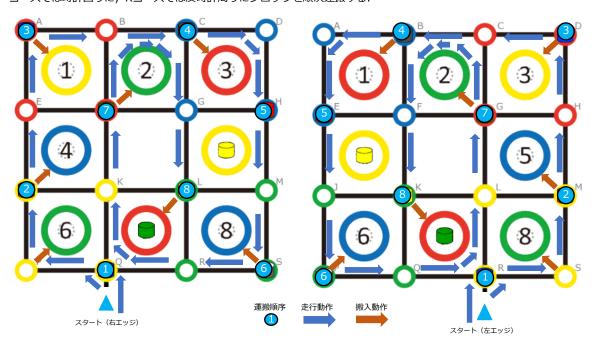


図2.2.2 交点サークル上における直進・旋回動作の例示

設計モデル



3.1 設計意図・方針

1.3および2.2より設計意図・方針は表3.1.1に示す.

表3.1.1 設計意図・方針

要件	意図	設計方針
来年度以降も同じコード・モデルが 使用できる	開発時間短縮のため、GitHubを使用して誰でもすぐに開発できるようにしたい、 初めて参加するメンバにもモデルを使用して 説明することによって教育時間短縮をしたい なるべくすぐに開発に取り組める環境を提供 し、メンバのやる気を維持させたい	走行体システムの責務を状態管理と動作 モデル・判別モデル・デバイスドライバ に分ける
機能の追加・変更に対応できる	後から新しい要素や戦略を追加するとき走行 体システムの変更箇所を少なくしたい	パッケージごとの関係において相互参照 しないように誘導可能性を一方向にする

3.2 構造設計(全体)

3.1の設計意図・方針より、走行体システムの全体像を図3.2.1に示す。また、各パッケージの責務を表3.2.1に示す。

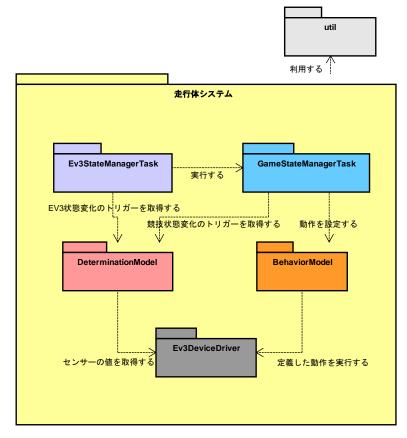


図3.2.1 走行体システムの全体像

表3.2.1 走行体システムの全体像

配置	配置 パッケージ名 責務	
走行体システム	Ev3StateManager	走行体の3状態(「ゲーム開始前」・「ゲーム中」・「ゲーム終了」)を管理する
	GameStateManager	3競技「走行競技」・「ブロックビンゴ競技」・「ガレージ競技」を管理する
	DeterminationModel	2.2で定義した検知条件から状態の変化や動作モデルの終了条件について判定する
	BehaviorModel	2.2で定義した動作から走行体の動作を生成する
	Ev3DeviceDriver	EV3APIを使用して、ハードウェアへのアクセスを抽象化する
その他	util	共通で使う機能を管理する

3.3 振舞設計(全体)

3.1, 3.2より走行体システムにおけるタスク管理について表3.3.1に示す.

表3.3.1 走行体システムのタスク管理

タスク	周期	優先度	責務と設計意図	実行パッケージ
EV3_STATE_ MANAGER_TASK	200ms	高	走行体の3状態を管理するタスク. 走行体はいつでも止められる必要があるため,優先度を一番高く設定 している.	走行体の状態管理 判定モデル
GAME_STATE_ MANAGER_TASK	10ms	中	3つの競技区間を管理するタスク. 走行体の状態管理によってこのタスクは開始・停止をするため,走行 体の状態管理タスクより低めの優先度を設定している.	競技区間の状態管理 判定モデル 動作モデル

4 設計モデル

3.4 構造設計(詳細)

図3.2.1における走行体システムの詳細構造を図3.4.1に示す. 走行システムの詳細構造にはブロックビンゴ競技に必要と考えた部分のみを記載している. また, EV3APIやutilパッケージなども省略している.

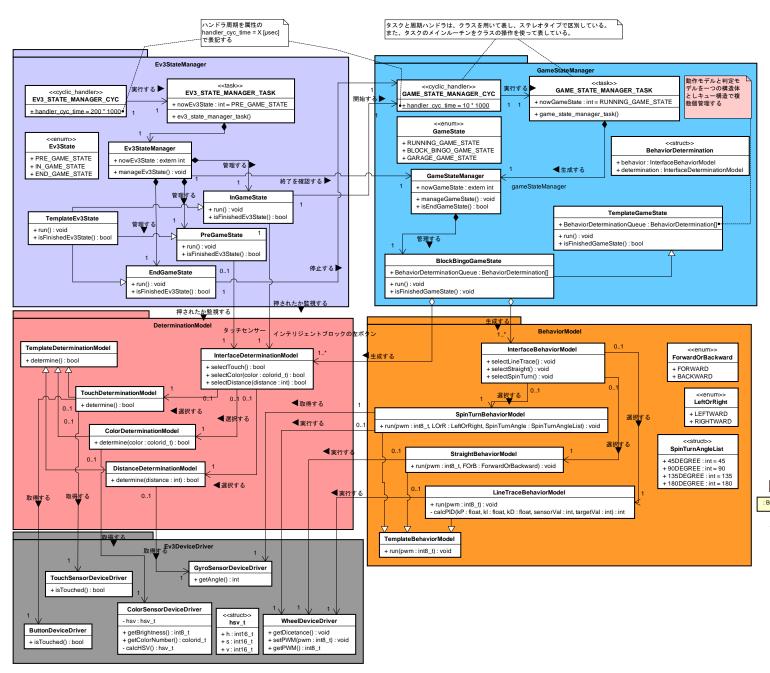


図3.4.1 走行体システムの詳細設計

3.5 振舞設計(詳細)

図3.4.1における走行体システムの振舞を図3.5.1、図3.5.2に示す。図3.5.1は、ブロックビンゴ競技に着目した全体の振舞を示している。図3.5.1は、具体的な動作の一例を例示している。

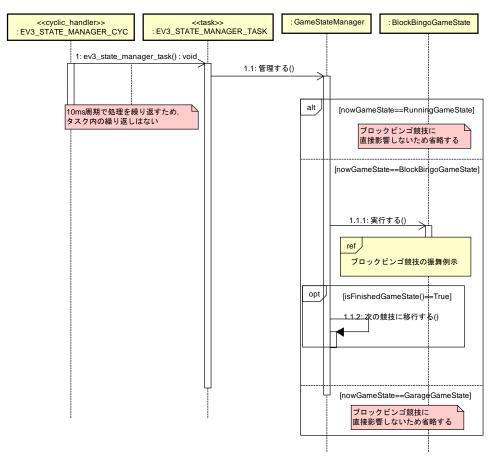


図3.5.1 ブロックビンゴ競技に着目したシステム全体の振舞

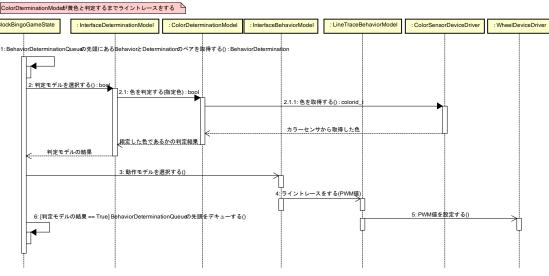


図3.5.2 ブロックビンゴ競技の振舞例示