

Champlify

Character pattern formation generator powered by Amplify

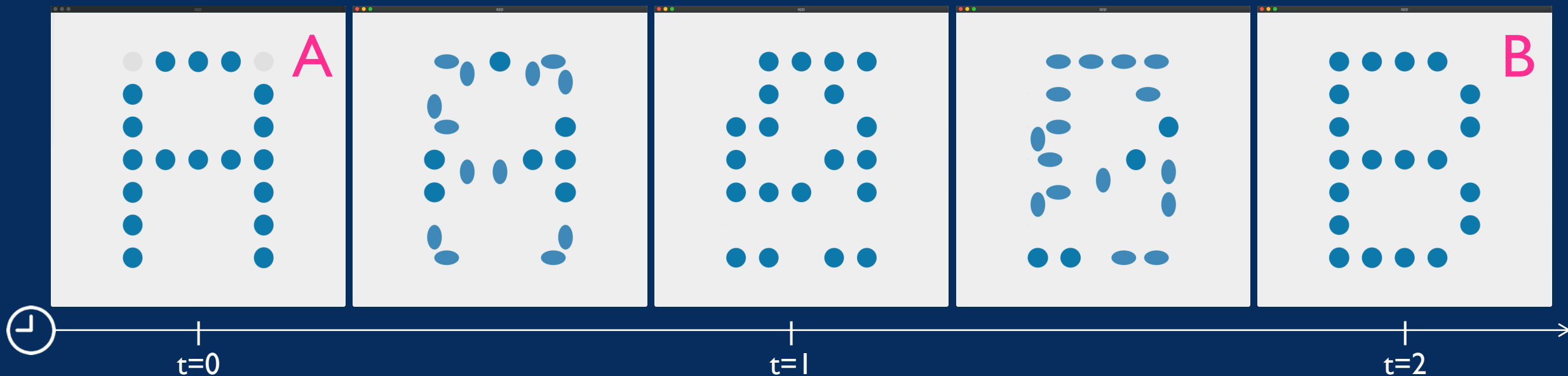
奥村 圭祐

<https://github.com/KeiI8/champlify>

Fixstars Amplify Hackathon

Mar. 28th 2021

作ったもの






エージェント(青点)によって構成された
任意のアルファベット2文字間を最短時間(aka. メイクスパン)で遷移

ターゲット割当 & 経路計画を同時に解く最適化問題
(unlabeled-MAPF; Multi-Agent Path Finding)

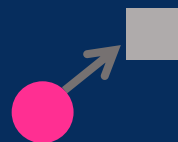
Unlabeled-MAPF

given

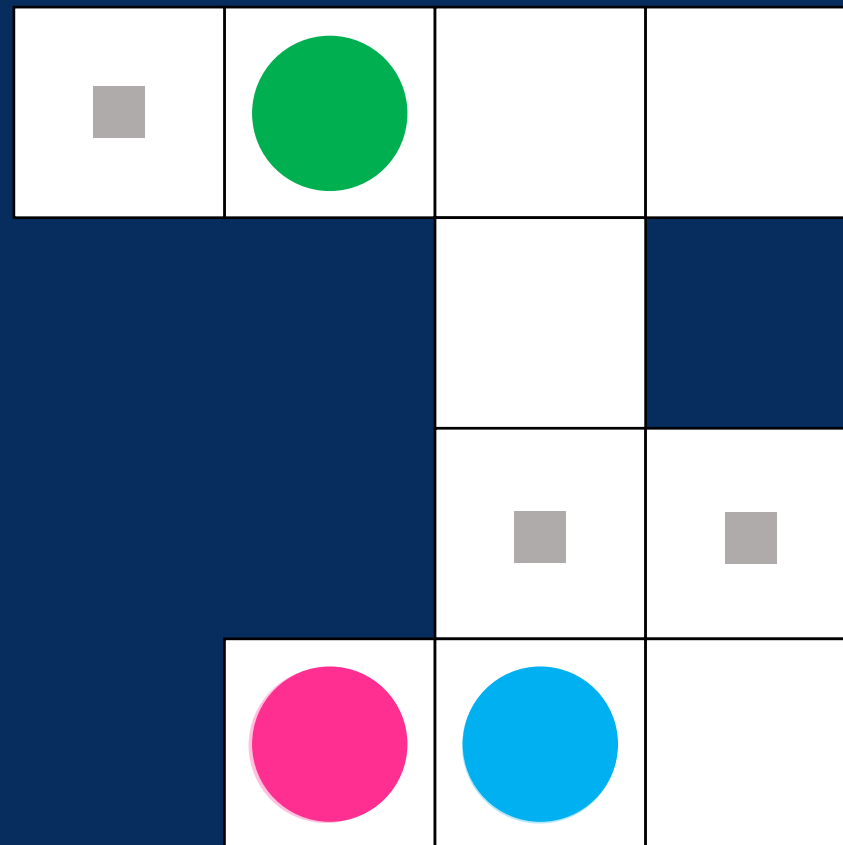
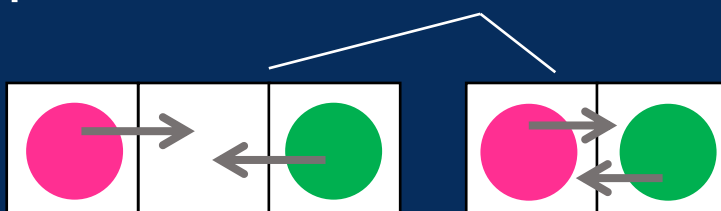
-  graph
-  agents (starts)
-  targets

solution

target assignment



paths without collisions



objective

minimize last arrival timestep (aka. makespan)

*時間は離散化, 必ず実行可能解が存在

応用例



YouTube/Mind Blowing Videos



Twitter/@knaohiro I



YouTube/StarCraft

自動倉庫の荷物運搬, ロボットサッカー, デジタルゲームなど種々様々
(かなり実用的な問題)

アプリケーションの構成



最適化の方針

メイクスパン T (初期値 l) に対して実行可能解を求める

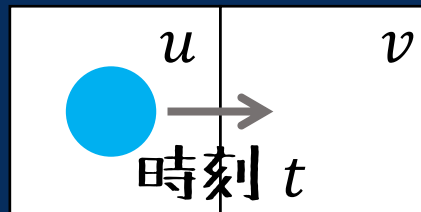
Success

メイクスパン最適な解が得られた, 後処理して終了

Fail

T をインクリメントしてやり直し

定式化 - 変数



$x_{(u,v),t}$: 辺 (u, v) を時刻 $[t, t + 1]$ で使用するかの二値数

定式化 - 制約条件

初期位置から出発 $\sum_{v \in \partial(s)} x_{(s,v),0} = 1$ $\forall s \in \text{スタートの集合}$

ターゲットへの到達 $\sum_{u \in \partial(g)} x_{(u,g),T-1} = 1$ $\forall g \in \text{ターゲットの集合}$

頂点から出ていく数の制限 $\sum_{v \in \partial(u)} x_{(u,v),t} \leq 1$ $\forall u \in V, 0 \leq t < T$

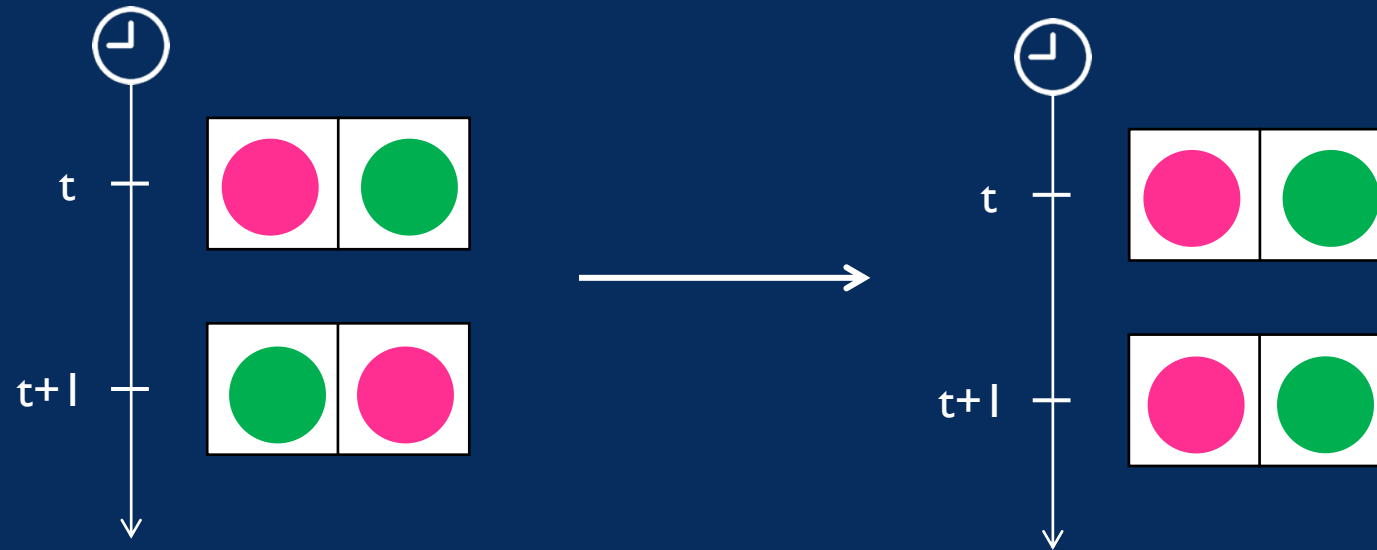
頂点に入ってくる数の制限 $\sum_{u \in \partial(v)} x_{(u,v),t} \leq 1$ $\forall v \in V, 0 \leq t < T$

経路の連続性 $\sum_{u \in \partial(v)} x_{(u,v),t} = \sum_{u \in \partial(v)} x_{(v,u),t+1}$ $\forall v \in V, 0 \leq t < T - 1$

$$\partial(u) \stackrel{\text{def}}{=} \{v \mid (u,v) \in E\} \cup \{u\}$$

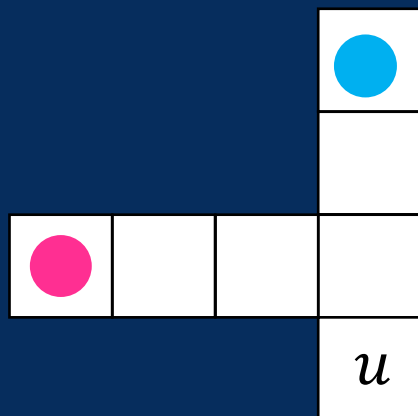
解の後処理

制約条件は辺上での衝突回避を課していないので後処理が必要



小テク - 枝刈り

{スタートから / ターゲットに} 到達不可能な (頂点, 時刻) のペアは事前に検出
問題を簡易化しておく



$t \leq 2$ では誰も u に辿り着けない
(幅優先探索x1で求められる)



$$x_{(u,v),t} = 0 \quad v \in \partial(u), 0 \leq t \leq 2$$

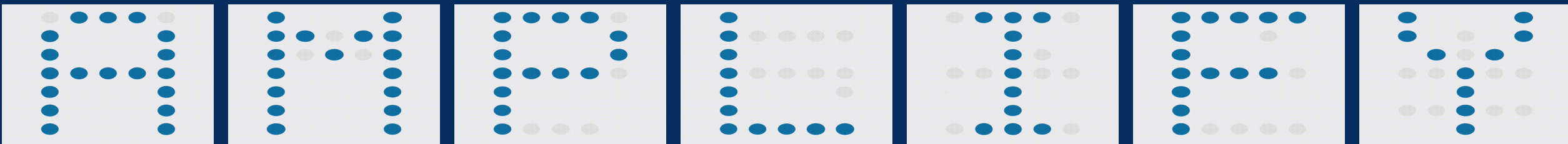
$$x_{(v,u),t} = 0 \quad v \in \partial(u), 0 \leq t < 2$$

ターゲットに到達できない (頂点, 時刻) も同様に検出可

Take-home Message

コア技術 ターゲット割当 & 経路計画の同時最適化

作ったもの 文字遷移の(リアルタイム)アニメーション生成



感想など

個人的なモチベ

手軽に使えるような組合せ最適化ソルバを試しておきたかった
(Gurobi や Google OR-Tools あたりは使用経験あり, Amplify は知りませんでした)

定式化してから実装までがスムーズにできた印象

手法の弱点

API を頻繁に叩くことが前提なので制限が怖い
大きいインスタンスは厳しい (e.g., 1000 agents)

その他

最初は (頂点, 時刻) を変数にしようとしたが駄目そう
js – python 間の非同期処理はちゃんとやっていない... もっとヌルヌル動かしたい
いい応用例あったら教えてください！