



Grupo 1

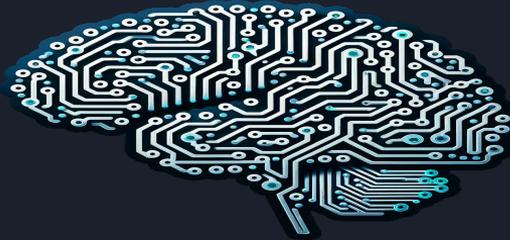
Scanalyzer 3D (IPC 2011)

- * Eduardo Silva Waski
- * João Paulo
- * Giovanni
- * José Victor
- * Mateus
- * Giovana Martins de Brito
- * Julio
- * Christian Hirsch Santos
- * Fabio
- * Arthur Heleno do Couto da Silva

Fundamentos Lógicos da IA - Trabalho 1

Sobre a competição que participou:

A **Competição Internacional de Planejamento** é uma competição científica voltada para avaliar e comparar o desempenho de algoritmos de planejamento automático.



O domínio **Scanalyzer 3D** foi um dos domínios de planejamento usados na **sétima Competição Internacional de Planejamento (IPC-7)** em 2011. Se destacou como um dos domínios mais complexos da competição.

Sobre os autores:



Dr. Malte Helmert:

- Especialidade em resolução de problemas e planejamento automatizado

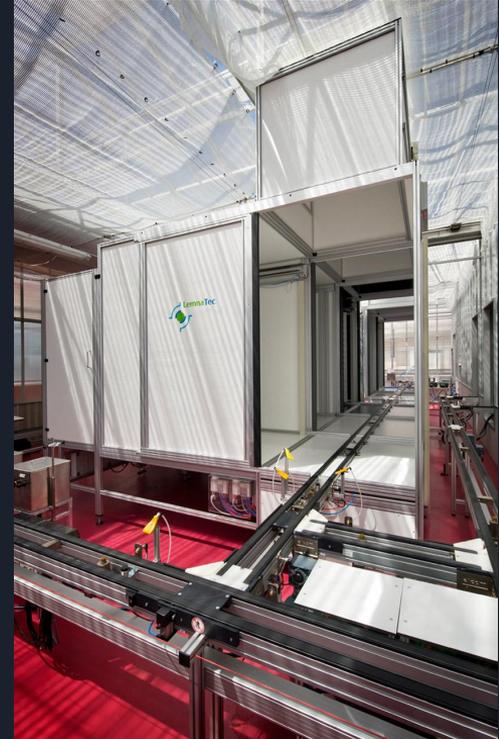
Hauke Lasinger:

- Profissional de uma empresa alemã especializada em soluções de fenotipagem automatizada e análise de dados para pesquisa científica e agricultura.

Sobre os autores: Qual a motivação do domínio existir

Ajudar a movimentar plantas em smart greenhouses (estufas inteligentes).

Devido ao espaço limitado e para diminuir custos, é necessário utilizar esteiras longas invés de várias mais curtas.



Sobre os autores: Qual a motivação do domínio existir

Outra dificuldade é a necessidade de colocar o máximo de plantas possível nas esteiras, o que significa muitas movimentações de várias plantas ao mesmo tempo.



Sobre os autores: Qual a motivação do domínio existir

As smart greenhouses podem ser muito grandes. A exemplo disso, na Universidade de Adelaide existem esteiras de mais de 1km de extensão e feitas para passar mais de 2400 plantas diariamente.





Sobre os autores: Qual a motivação do domínio existir

Em resumo, o domínio Scanalyzer existe para:

1. Aumentar a eficiência na coleta de dados fenotípicos em estufas.
2. Facilitar a automação de processos logísticos complexos.
3. Proporcionar um campo de pesquisa para o desenvolvimento de algoritmos de planejamento que podem ser aplicados em contextos reais.

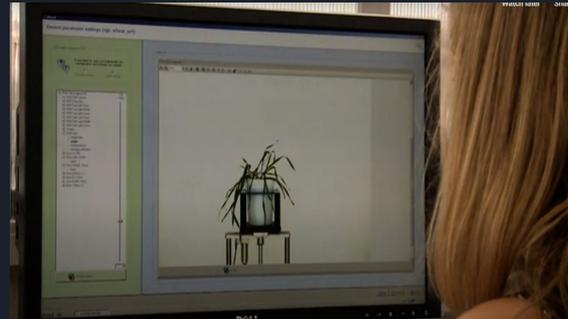
Do que o domínio se trata

- Análise crítica sobre o domínio e o conjunto de problemas

O domínio trata sobre a Logística de Estufa e como organizar o transporte de plantas em esteiras



Para serem enviadas para análise fenotípica e serem retornadas a um local determinado



Do que o domínio se trata

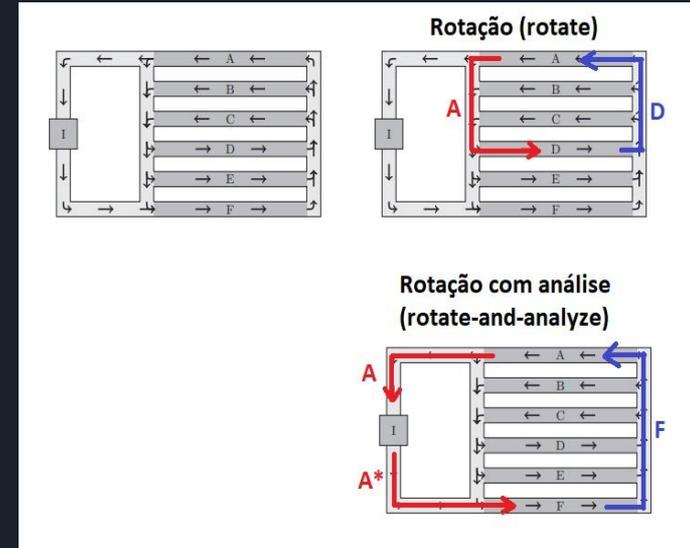
-Plantas são separadas em grupos em cada esteira

-As plantas de um grupo não podem ser separadas

-2 tipos de movimentos estão disponíveis :

Rotação - Grupo troca de posição

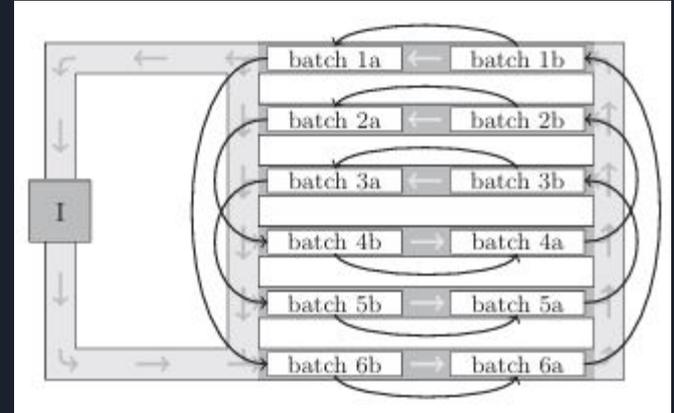
Rotação com análise - Grupo troca de posição passando por uma máquina de análise



Do que o domínio se trata

O domínio admite a subdivisão dos grupos, porém não trata de divisões individuais das plantas

O domínio conta também com um sistema de custo de ações para a otimização do problema





Ferramenta de apoio

Planimation

Quickly build an visualisation animation form problem or VFG file shows the plan and subplan for each problem.

Problem

Build visualisation from
problem

EXPLORE

VFG

Build visualisation from
VFG

EXPLORE

User Manual

Source document and
user manual

EXPLORE

Demo

Demo File for
visualisation

EXPLORE

Explicação dos Predicados

```
(define (domain scanalyzer3d)
  (:requirements :typing :action-costs)
  (:types segment car - object)
  (:predicates (on ?c - car ?s - segment)
    (analyzed ?c - car)
    (CYCLE-2 ?s1 ?s2 - segment)
    (CYCLE-4 ?s1 ?s2 ?s3 ?s4 - segment)
    (CYCLE-2-WITH-ANALYSIS ?s1 ?s2 - segment)
    (CYCLE-4-WITH-ANALYSIS ?s1 ?s2 ?s3 ?s4 - segment))
  (:functions (total-cost) - number))
```

Predicado 4: existe um ciclo entre quatro esteiras s1, s2, s3 e s4

Predicado 5: existe um ciclo com análise entre duas esteiras s1 e s2

Predicado 6: existe um ciclo com análise entre quatro esteiras s1, s2, s3 e s4

Nome do Domínio:
scanalyzer3d

Requisitos:
tipagem (quais os tipos) e custo de ações (específica que cada ação possui um valor de custo)

Tipos: esteira (s) e carro (c)

Predicado 1: um carro c está em uma esteira s

Predicado 2: um carro c foi analisado

Predicado 3: existe um ciclo entre duas esteiras s1 e s2

Ação de Análise 2

```
(:action analyze-2
:parameters (?s1 ?s2 - segment ?c1 ?c2 - car)
:precondition (and (CYCLE-2-WITH-ANALYSIS ?s1 ?s2)
  (on ?c1 ?s1)
  (on ?c2 ?s2)
)
:effect (and (not (on ?c1 ?s1))
  (not (on ?c2 ?s2))
  (on ?c1 ?s2)
  (on ?c2 ?s1)
  (analyzed ?c1)
  (increase (total-cost) 3))
)
```

Nome da Ação: analyze-2
uma análise envolvendo 2
pares de esteiras e carros

Parâmetros: duas esteiras
s1 e s2 e dois carros c1 e c2

Pré condição: existe um ciclo
com análise entre s1 e s2

E o carro c1 está na esteira s1

E o carro c2 está na esteira s2

E o carro c1 foi analisado

E o custo total foi
incrementado em 3

Efeito: o
carro c1 não
está mais na
esteira s1

E o carro c2 não está
mais na esteira s2

E o carro c1 agora está
na esteira s2

E o carro c2 agora
está na esteira s1

Ação de Análise 4

Nome da Ação: analyze-4
uma análise envolvendo 4 pares de esteiras e carros

Parâmetros: quatro esteiras s1, s2, s3 e s4 e quatro carros c1, c2, c3 e c4

```
(:action analyze-4
:parameters (?s1 ?s2 ?s3 ?s4 - segment ?c1 ?c2 ?c3 ?c4 - car)
:precondition (and (CYCLE-4-WITH-ANALYSIS ?s1 ?s2 ?s3 ?s4)
  (on ?c1 ?s1)
  (on ?c2 ?s2)
  (on ?c3 ?s3)
  (on ?c4 ?s4)
)
:effect (and (not (on ?c1 ?s1))
  (not (on ?c2 ?s2))
  (not (on ?c3 ?s3))
  (not (on ?c4 ?s4))
  (on ?c1 ?s4)
  (on ?c2 ?s1)
  (on ?c3 ?s2)
  (on ?c4 ?s3)
  (analyzed ?c1)
  (increase (total-cost) 3)
)
)
```

Pré condição: existe um ciclo com análise entre as esteiras s1, s2, s3 e s4

E o carro c1 está na esteira s1
E o carro c2 está na esteira s2
E o carro c3 está na esteira s3
E o carro c4 está na esteira s4

Efeito: o carro c1 não está mais na esteira s1
E o carro c2 não está mais na esteira s2
E o carro c3 não está mais na esteira s3
E o carro c4 não está mais na esteira s4
E o carro c1 agora está na esteira s4
E o carro c2 agora está na esteira s1
E o carro c3 agora está na esteira s2
E o carro c4 agora está na esteira s3
E o carro c1 foi analisado
E o custo total foi incrementado em 3

Ação de Rotação 2

```
(:action rotate-2
:parameters (?s1 ?s2 - segment ?c1 ?c2 - car)
:precondition (and (CYCLE-2 ?s1 ?s2)
                  (on ?c1 ?s1)
                  (on ?c2 ?s2)
                  )
:effect (and (not (on ?c1 ?s1))
             (not (on ?c2 ?s2))
             (on ?c1 ?s2)
             (on ?c2 ?s1)
             (increase (total-cost) 1)
            )
)
```

Nome da ação: rotate-2

uma rotação envolvendo dois pares de esteiras e carros

Parâmetros: duas esteiras s1 e s2 e dois carros c1 e c2

Pré condição: existe um ciclo entre as esteiras s1 e s2

E o carro c1 está na esteira s1

E o carro c2 está na esteira s2

Efeito: o carro c1 não está mais na esteira s1

E o carro c2 não está mais na esteira s2

E o carro c1 agora está na esteira s2

E o carro c2 agora está na esteira s1

E o custo total foi incrementado em 1

Ação de Rotação 4

Nome da ação :rotate-4
uma rotação envolvendo quatro pares de esteiras e carros

Parâmetros: quatro esteiras s1, s2, s3 e s4 e quatro carros c1, c2, c3 e c4

Pré condição: existe um ciclo entre as esteiras s1, s2, s3 e s4
E o carro c1 está na esteira s1
E o carro c2 está na esteira s2
E o carro c3 está na esteira s3
E o carro c4 está na esteira s4

```
(:action rotate-4
:parameters (?s1 ?s2 ?s3 ?s4 - segment ?c1 ?c2 ?c3 ?c4 - car)
:precondition (and (CYCLE-4 ?s1 ?s2 ?s3 ?s4)
  (on ?c1 ?s1)
  (on ?c2 ?s2)
  (on ?c3 ?s3)
  (on ?c4 ?s4)
)
:effect (and (not (on ?c1 ?s1))
  (not (on ?c2 ?s2))
  (not (on ?c3 ?s3))
  (not (on ?c4 ?s4))
  (on ?c1 ?s4)
  (on ?c2 ?s1)
  (on ?c3 ?s2)
  (on ?c4 ?s3)
  (increase (total-cost) 1)
)
)
```

Efeito: o carro c1 não está mais na esteira s1
E o carro c2 não está mais na esteira s2
E o carro c3 não está mais na esteira s3
E o carro c4 não está mais na esteira s4
E o carro c1 agora está na esteira s4
E o carro c2 agora está na esteira s1
E o carro c3 agora está na esteira s2
E o carro c4 agora está na esteira s3
E o custo total foi incrementado em 1

CUSTOS de cada ação

```
(:functions (total-cost) - number)
```



:function - permite definir uma variável que armazena valores numéricos



total-cost - variável que armazena o valor de cada ação

Custos de cada ação

Ações: analyze-2 e analyze-4

(:action analyze-2

```
:parameters (?s1 ?s2 - segment ?c1 ?c2 - car)
:precondition (and (CYCLE-2-WITH-ANALYSIS ?s1 ?s2)
                  (on ?c1 ?s1)
                  (on ?c2 ?s2)
                  )
:effect (and (not (on ?c1 ?s1))
             (not (on ?c2 ?s2))
             (on ?c1 ?s2)
             (on ?c2 ?s1)
             (analyzed ?c1)
             (increase (total-cost) 3))
```

(:action analyze-4

```
:parameters (?s1 ?s2 ?s3 ?s4 - segment ?c1 ?c2 ?c3 ?c4 - car)
:precondition (and (CYCLE-4-WITH-ANALYSIS ?s1 ?s2 ?s3 ?s4)
                  (on ?c1 ?s1)
                  (on ?c2 ?s2)
                  (on ?c3 ?s3)
                  (on ?c4 ?s4)
                  )
:effect (and (not (on ?c1 ?s1))
             (not (on ?c2 ?s2))
             (not (on ?c3 ?s3))
             (not (on ?c4 ?s4))
             (on ?c1 ?s4)
             (on ?c2 ?s1)
             (on ?c3 ?s2)
             (on ?c4 ?s3)
             (analyzed ?c1)
             (increase (total-cost) 3))
)
```

Custo: +3

Tabela de execução dos problemas disponíveis

fast-downward Planejador	Solução(Cost)		
	Tempo max: 5min		
Instâncias	OPTIMAL(OPT) seq-opt-fdss-2023	SATISFICING(SAT) seq-sat-fdss-2023	AGILE(AGL)
1	18	Sem Solução	24
2	22	Sem Solução	26
3	26	26	38
4	24	Time Limit	32
5	30	Time Limit	36
6	36	Time Limit	54
7	30	Time Limit	42
8	Time Limit	Time Limit	46
9	Time Limit	Time Limit	70
10	Time Limit	Time Limit	50
11	Time Limit	Time Limit	56
12	Time Limit	Time Limit	86
13	Time Limite	Time Limit	60
14	Time Limit	Time Limit	66
15	Time Limit	Time Limit	102
16	Time Limit	Time Limit	68
17	Time Limit	Time Limit	76
18	Time Limit	Time Limit	118
19	Time Limit	Time Limit	78
20	Time Limit	Time Limit	86
21	Time Limit	Time Limit	134
22	13	Sem Solução	13
23	13	Sem Solução	13
24	13	Sem Solução	13
25	26	Time Limit	26
26	30	Time Limit	54
27	34	Time Limit	62
28	Time Limit	Time Limit	67
29	Time Limit	Time Limit	99
30	Time Limit	Time Limit	119



Número de soluções encontradas em cada configuração

Optimal (OPT): 13/30

Satisficing (SAT): 1/30

Agile (AGL): 30/30

Performance por Custo

