Goal-directed Generation of Exercise Sets for Upper-Limb Rehabilitation

José C. Pulido, José C. González,

Arturo González-Ferrer, Javier García, Fernando Fernández,

Antonio Bandera, Pablo Bustos and Cristina Suárez

WS 5: Knowledge Engineering for Planning and Scheduling

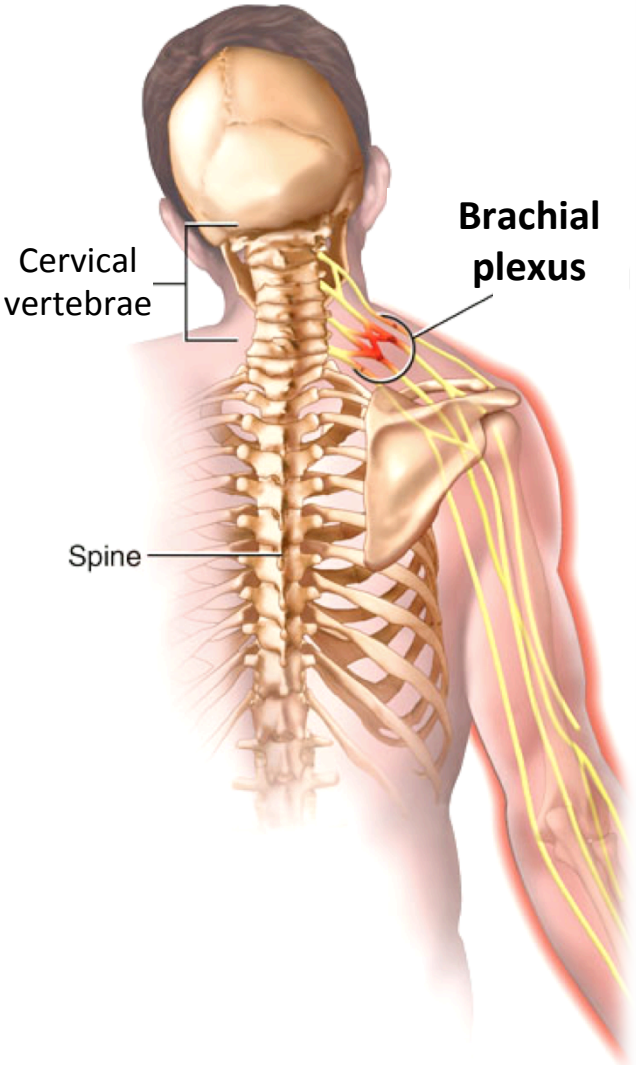
Overview

1. Introduction
2. Therapy Model
3. Comparison
   - Classical Planning
   - HTN Planning
4. Experiments
5. Discussion
6. Future work

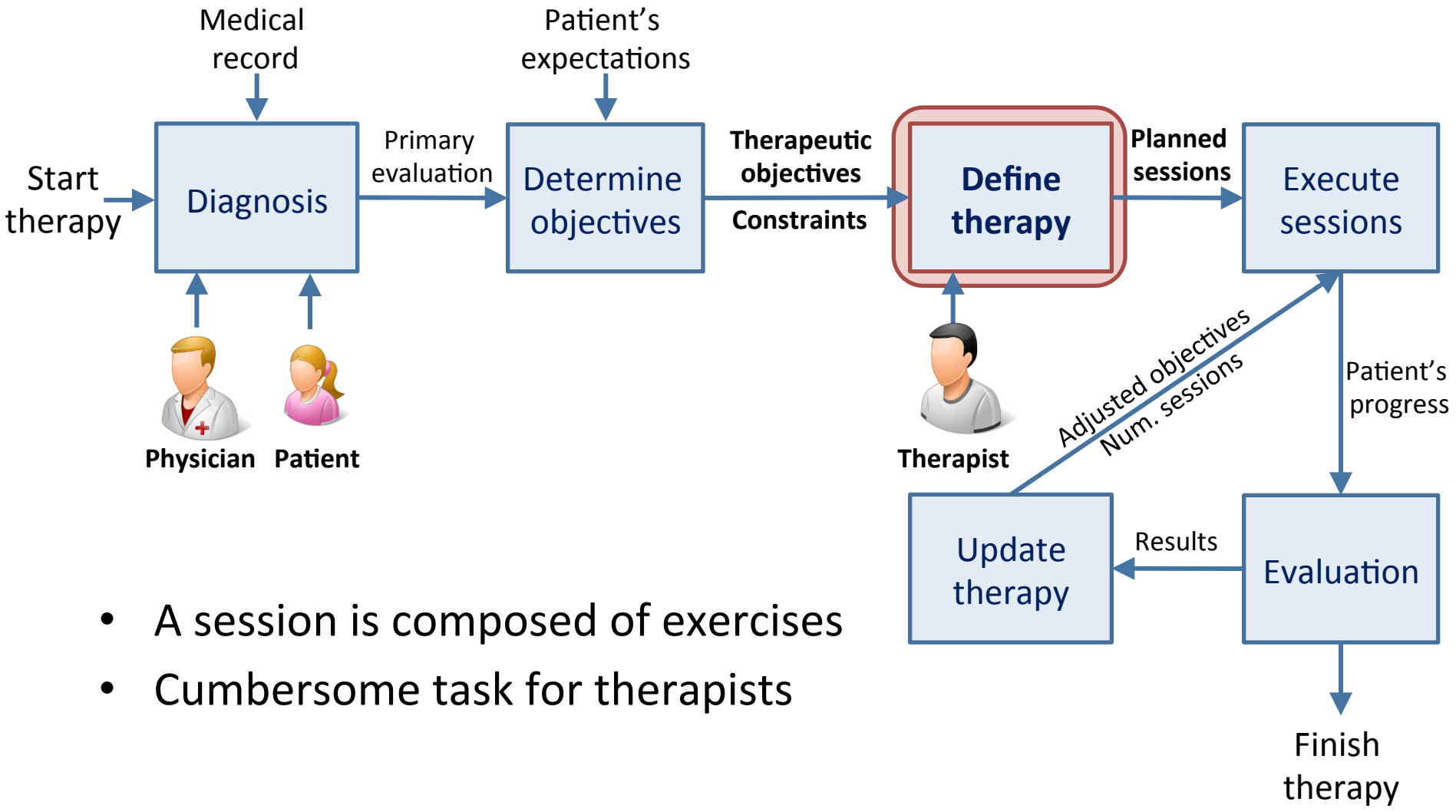
Background

- Obstetric Brachial Plexus Palsy
  - **Damage in nerves** around the shoulder
  - Loss of movement in the **upper limbs**
  - Requires **physical rehabilitation**

- Rehabilitation helps to:
  - ✓ Recover upper limb mobility
  - ✓ Reduce muscles rigidity
  - ✓ Increase patient's autonomy
    - Dressing
    - Eating



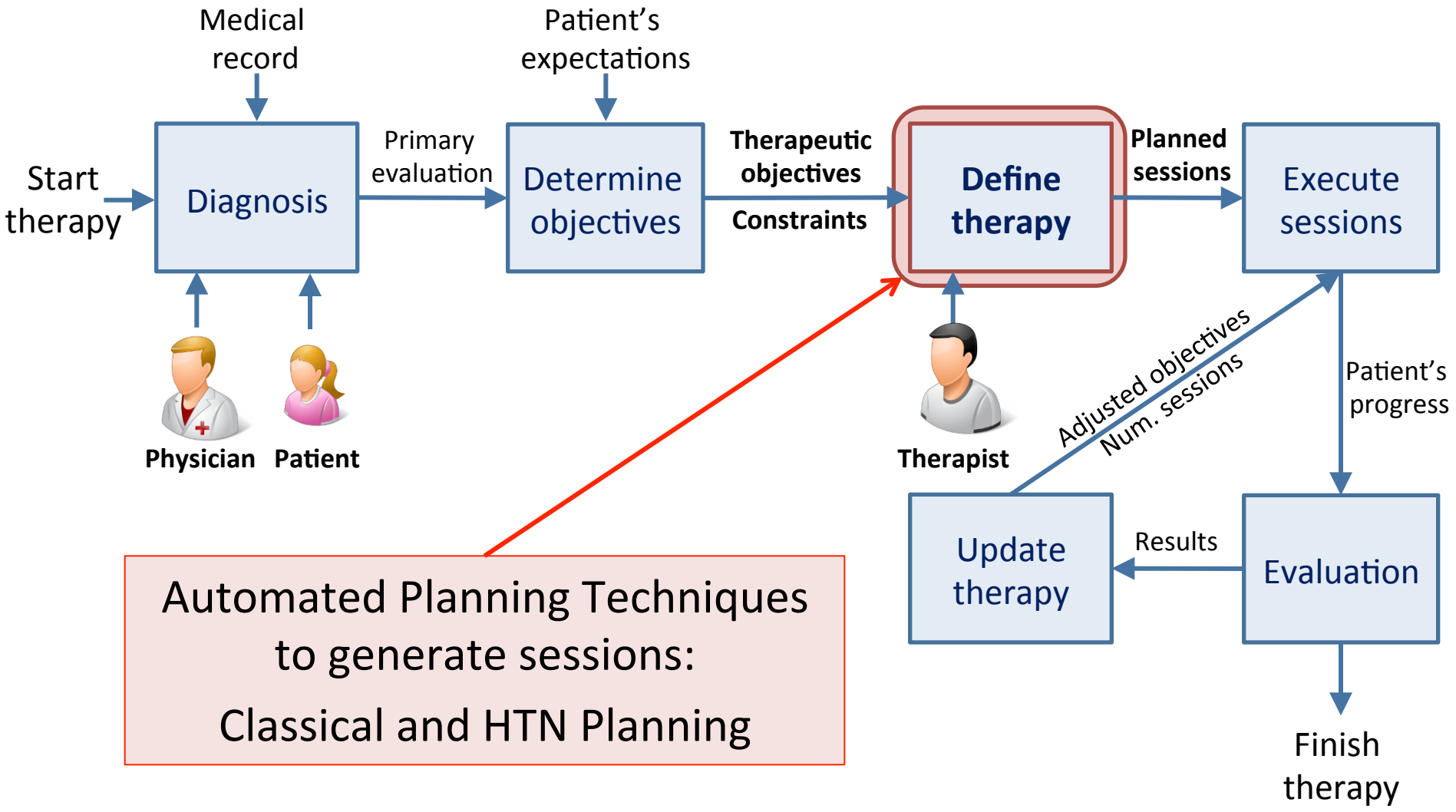
Rehabilitation Procedure



- A session is composed of exercises
- Cumbersome task for therapists

Img. source: Creative Can

Objectives



Img. source: Creative Can

Model

- TOCL (Therapeutic Objectives Cumulative Level)

Therapeutic Objectives	TOCLs
1. Bimanual	15
2. Fine unimanual	30
3. Coarse unimanual	5
4. Arm positioning	0
5. Hand positioning	0

- Each exercise has an **adequacy level** associated to each TOCL

- Objective: reaching the TOCLs
  - The sum of the adequacy levels of the planned exercises must **reach the respective TOCL** for each session

Model

- Exercises
  - Adequacy level for each therapeutic objective
  - Duration, intensity and difficulty
  - Group of exercise

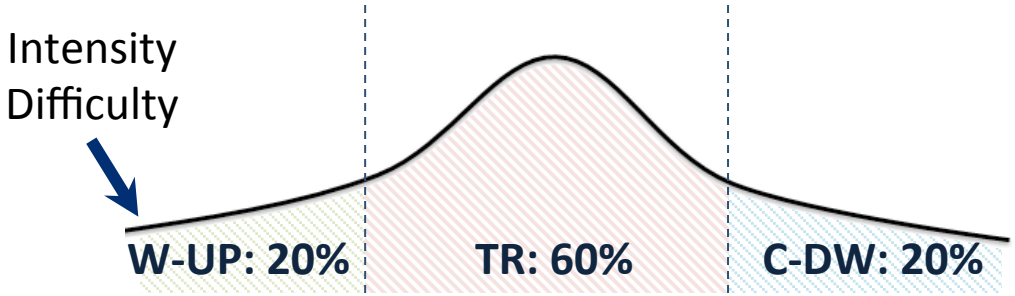
- For instance: “drawing figures”

Adequacy levels	
Bimanual	0
Fine unimanual	+3
Coarse unimanual	0
Arm positioning	0
Hand positioning	+2

Other Attributes	
Duration	3 min.
Intensity	Medium
Difficulty	High
Group of exercise	“handwritting”

Constraints

- A session has a max. and min. duration
  1. Warming up
  2. Training
  3. Cooling down



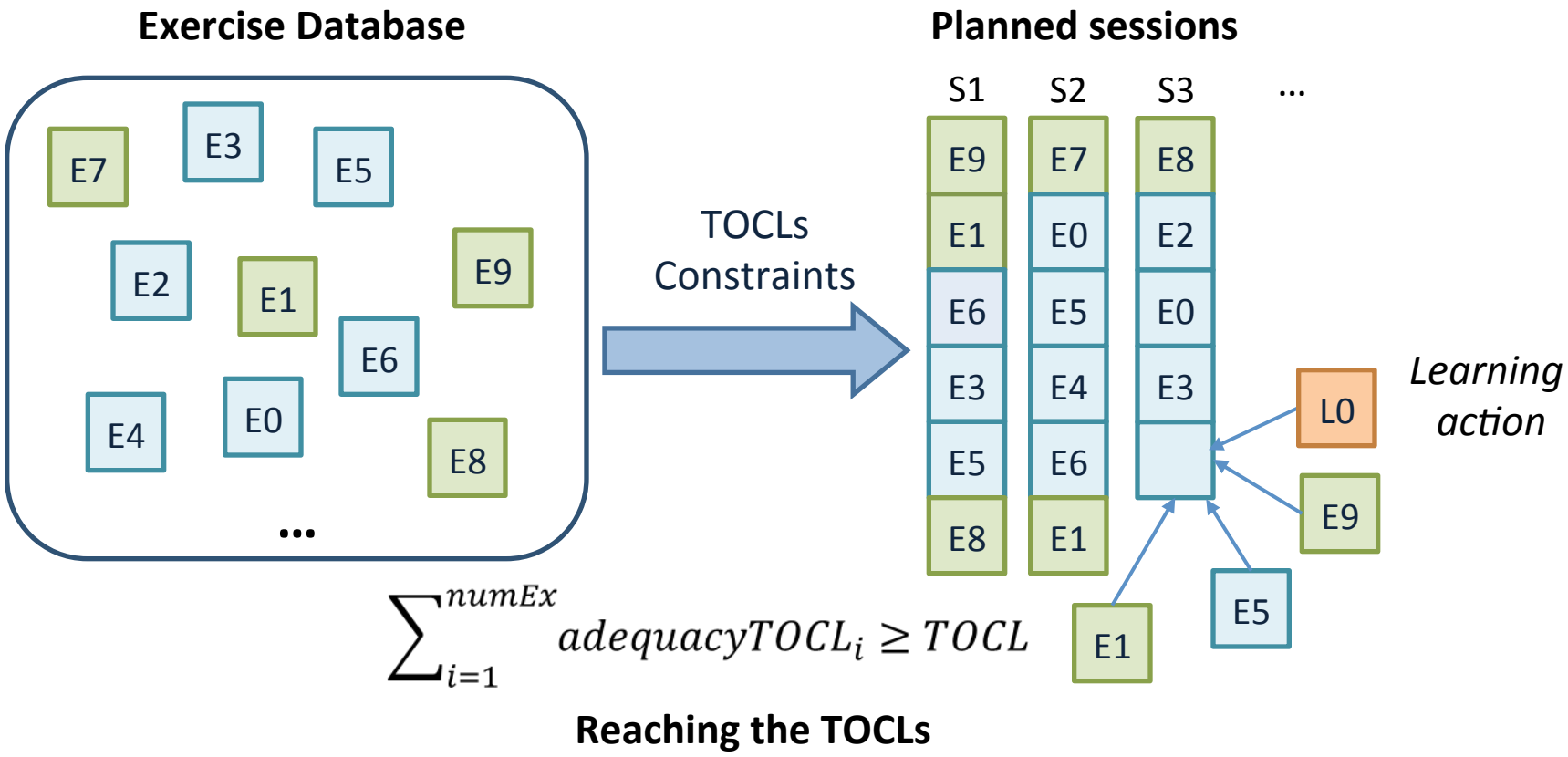
- Variability constraints
  - Exercises cannot reappear in one session
  - Distribution of exercises should be assorted throughout sessions
- Constraints of the patient
  - Avoid a certain group of exercises according to patient conditions

Learning (designing) new exercises

If there are no available exercises in the database, the planner can suggest to **learn a new exercise** to find a suitable plan.

- The attributes of the new exercise must allow to reach the TOCLs
- Previous knowledge is not needed
- Exercises from the database are preferred
- Therapists can "*teach*" new exercises

Planning Process



Classical Planning

- Domain and problem in PDDL
- Our model is based on:
  - Fluents and numeric preconditions
  - Action costs to control the exercise insertion
- We use CBP planner (Cost-Based Planner)

  (Fuentetaja et al. 2010)

Classical Planning

- Example plan with CBP-IPC2011 (first solution):

```
 0: (SESSION-START)
 1: (WARMUP-PHASE)
 2: (WARMUP-DATABASE-EXERCISE E0)
 3: (TRAINING-PHASE)
 4: (TRAINING-DATABASE-EXERCISE E11)
 5: (TRAINING-DATABASE-EXERCISE E12)
 6: (TRAINING-DATABASE-EXERCISE E10)
 7: (TRAINING-DATABASE-EXERCISE E9)
 8: (LEARN-TRAINING-EXERCISE
       T_SPATIAL_HAND A_MEDIUM D_LONG I_INTENSE)
 9: (COOLDOWN-PHASE)
10: (COOLDOWN-DATABASE-EXERCISE E15)
11: (SESSION-END)
```

Classical Planning

- Action for including exercises from database

```
(:action training-database-exercise
  :parameters (?e - exercise_training)
  :precondition (and
   (current_phase p_training)
   (< (p_duration p_training) (p_max_duration p_training))
   (> (- (session_index) (e_last_session ?e) ) 2)
   (not (= (exercise_index) (e_last_position ?e))))
  :effect (and
   (assign (e_last_session ?e) (session_index))
   (assign (e_last_position ?e) (exercise_index))
   (increase (p_duration p_training) (e_duration ?e))
   (increase (intensity_cumulative) (e_intensity ?e))
   (increase (difficulty_cumulative) (e_difficulty ?e))
   (increase (exercise_index) 1)
   (increase (TOCL t_bimanual) (e_adequacy ?e t_bimanual))
   (increase (TOCL t_unimanual_coarse) (e_adequacy ?e t_unimanual_coarse))
   (increase (TOCL t_unimanual_fine) (e_adequacy ?e t_unimanual_fine))
   (increase (TOCL t_spatial_arm) (e_adequacy ?e t_spatial_arm))
   (increase (TOCL t_spatial_hand) (e_adequacy ?e t_spatial_hand))))
```

Phase control

Variability

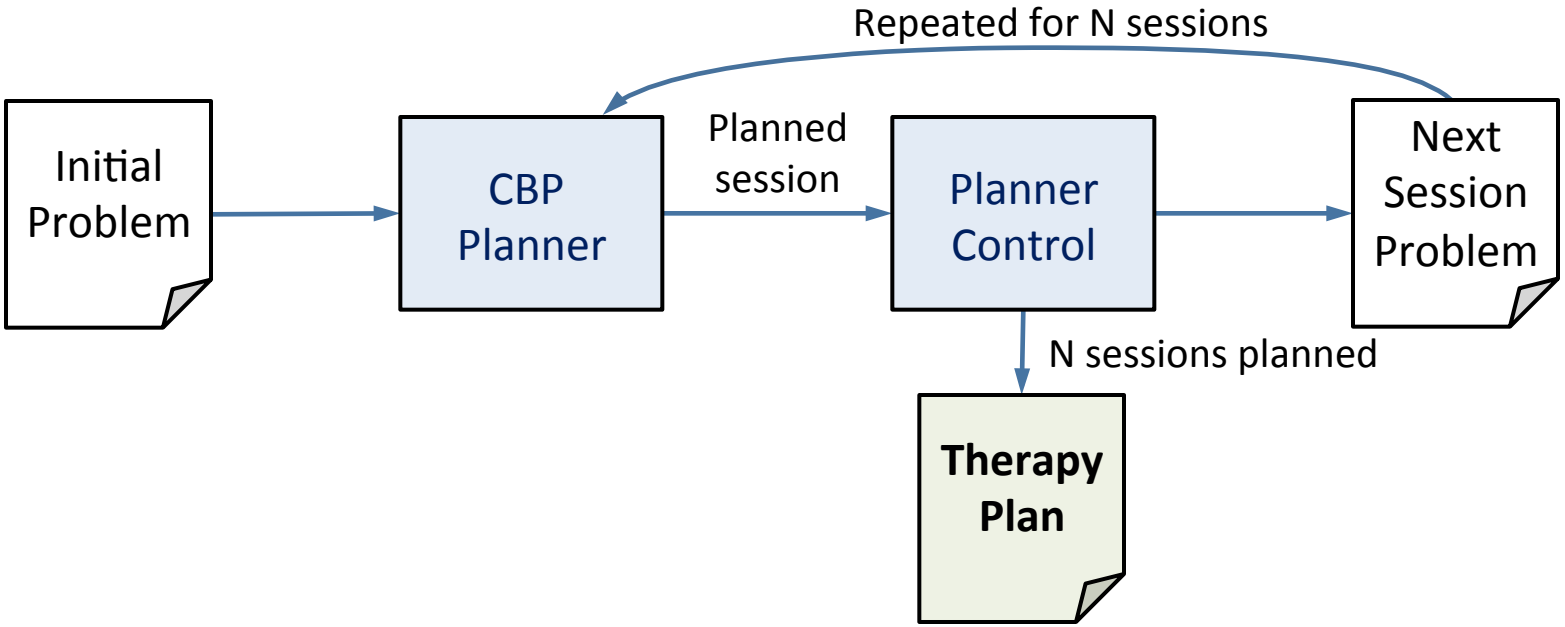
Classical Planning

- Action for **learning new exercises**
  - High cost to penalize overuse and favor exercise variability
  - The planner searches a suitable combination of the attributes of the exercise to reach the TOCLs

Classical Planning

- Due to the interaction between sessions, planning multiple sessions in one execution is much harder than planning a single session

- **Divide and Conquer** strategy (D&C)



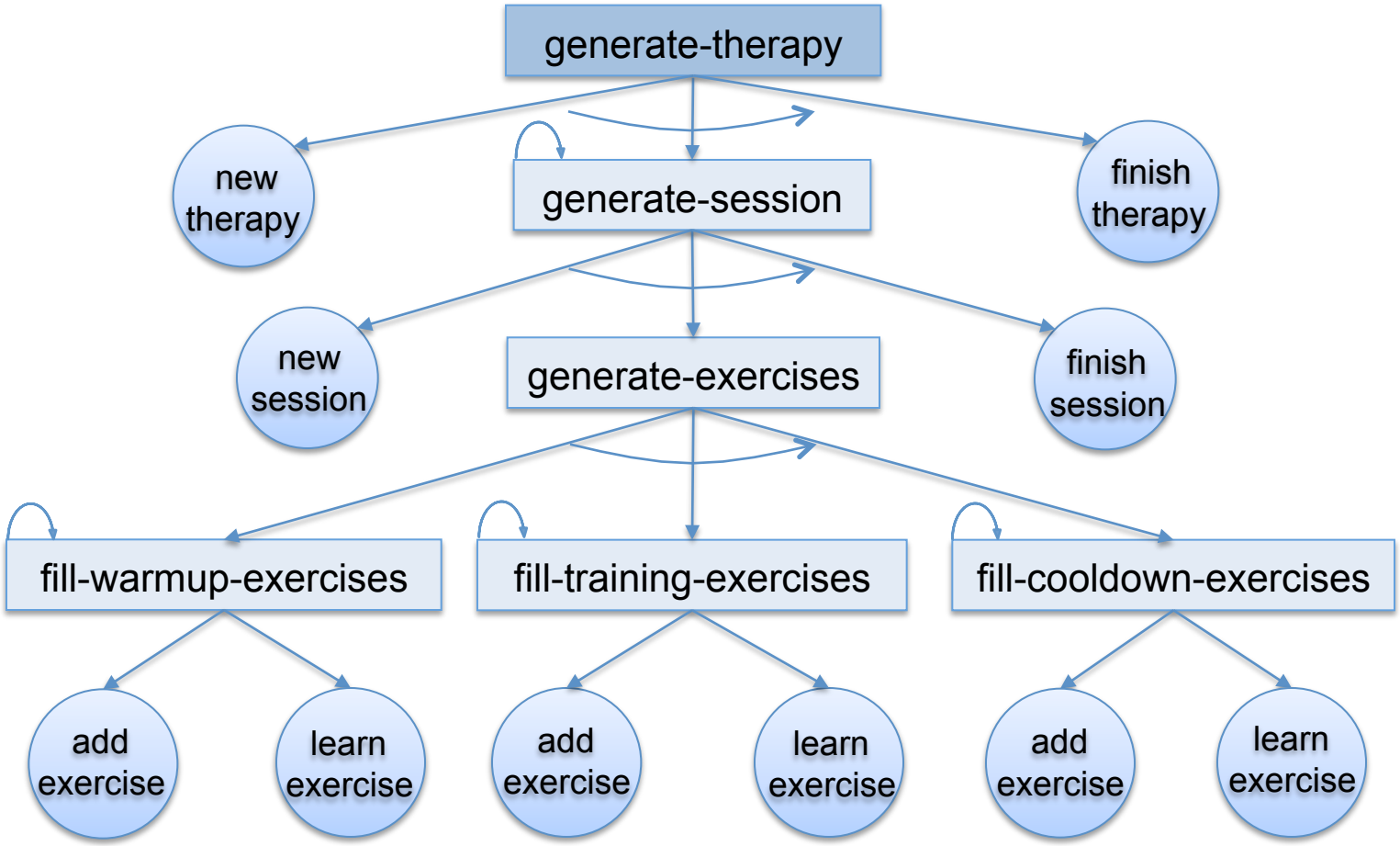
HTN Planning

- Hierarchical representation of the problem
- Extensible and configurable model to include human expert knowledge
- We use JSHOP2 (Simple Hierarchical Ordered Planner)

  (Nau et al. 2003)

HTN Planning

- HTN Model Schema



HTN Planning

- Planning Strategy:

Heuristic function of each exercise

$$ht_{ex} = \sum_{i=1}^{n_{objectives}} \left(\frac{1}{{d_i}^2 + 1} - \frac{ex_{times\_used}}{num_{sessions}} \right)$$

1 2

1

Contribution of the exercise to the TOCLs

- d_i is the difference between the desired TOCL and the cumulative level after including the exercise

2

Repetition penalty

- The higher the repetition, the higher the penalty

HTN Planning

```
;; Receives the session number                                   Task definition
(:method (fill-cooldown-exercises ?csn)
Method 1
   (:sort-by ?ht >)                                             Precondition 1
   ((e-target1 ?e ?et1)
   (current-acc t1 ?csn ?ct1a)
   (baseline t1 ?t1bl)
    ...
   (assign ?d1 (call - ?t1bl (call + ?et1 ?ct1a)))
    ...
   (assign ?h1 (call / 1 (call + (call * ?d1 ?d1) 1)))
    ...
   (e-used ?e ?n-used) (t-session-number ?tsn)
   (assign ?ht (call - (call + ?h1 ... ?h5) (call / ?n-used ?tsn)))
    ...
   (cooldown-time ?cst ?minST ?maxST)
   (cooldown-exercise ?e ?minST ?maxST)
   (not (used ?e ?csn))))
          ((!add-ex ?e cool-down)                               Actions and task calls
          (fill-cooldown-exercises ?csn))
Method 2
   (forall (?e) ((exercise ?e)) (used ?e ?csn))                 Precondition 2
          ((!learn))                                            Actions and task calls
Method 3
   ((current-session-time ?csn ?cst)                            Precondition 3
   (session-max-time ?csn ?maxST)
   (call <= ?cst ?maxST)
   (current-acc t1 ?csn ?ct1a) (TOCL t1 ?t1bl) (call >= ?ct1a ?t1bl)
    ...
          ((!finish-session ?csn)))                             Actions and task calls
```

Heuristic function

Phase control

Reaching TOCLs

Experiments

- Full therapy plan for 15 sessions using CBP

Planned exercises →

Sessions ↓

	1	2	3	4	5	6	7	8
1	e0	e9	e11	e12	e10	e7	e15	
2	e4	e2	e5	e6	**L**	**L**	**L**	e13
3	e1	e3	e8	**L**	**L**	**L**	**L**	e16
4	**L**	**L**	**L**	**L**	**L**	**L**	e17	
5	e0	e11	e12	e10	e9	**L**	e15	
6	e4	e2	e6	L19	e7	e5	L20	e13
7	e1	e3	L24	e8	L23	L22	e16	
8	L25	L26	L30	L27	L28	L29	e17	
9	e0	e12	e10	L31	e11	e9	e15	
10	e4	e2	L19	L20	e6	e7	e13	
11	e1	e3	L22	L23	L24	e8	e16	
12	**L**	L25	L26	L29	L30	L27	L28	e17
13	e0	e10	L21	e12	e9	e11	e15	
14	e4	e2	e7	e6	L20	L19	e13	
15	e1	e3	L24	e8	L22	L23	e16	

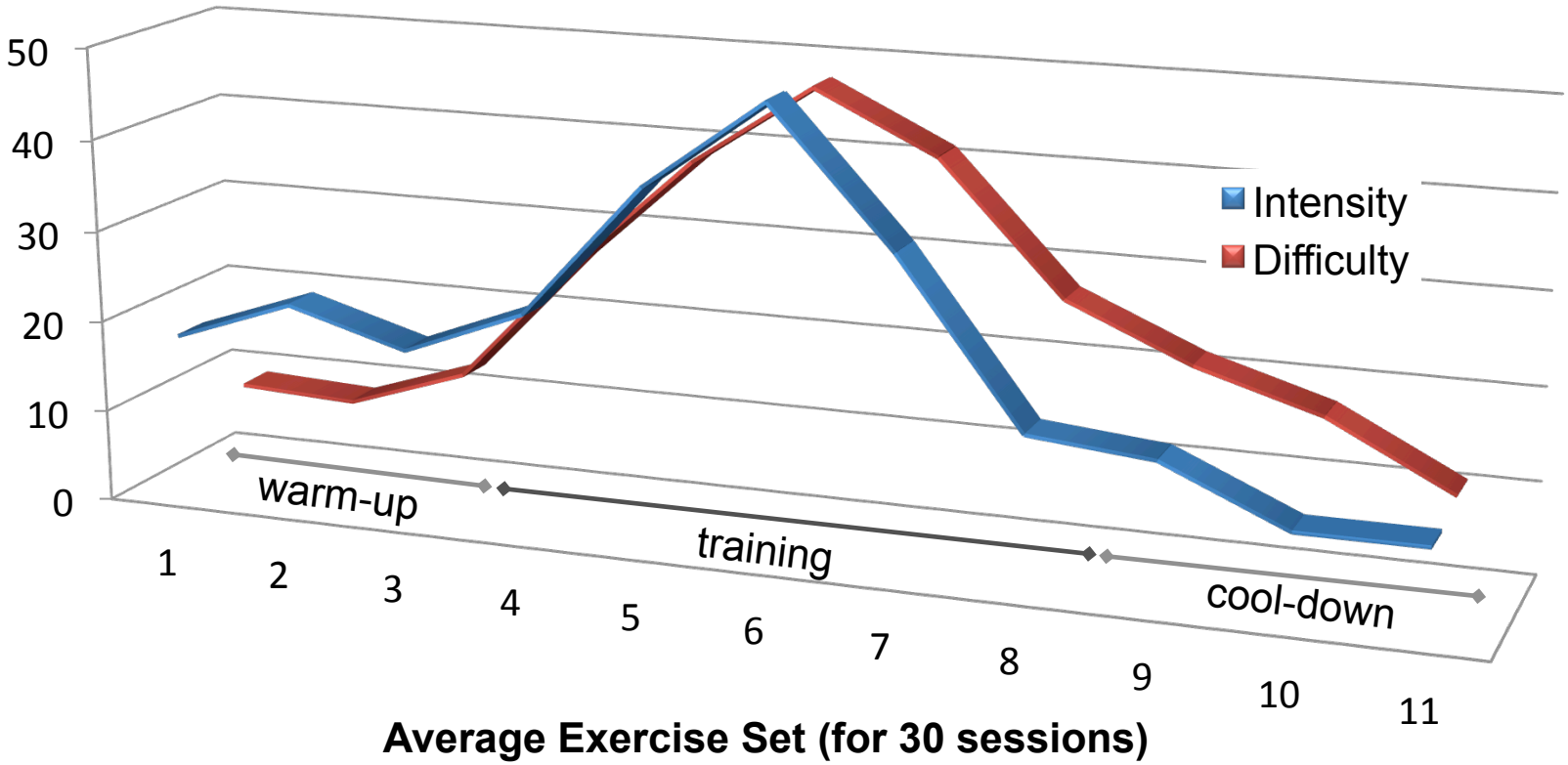
Few initial exercises

- Warm-up: 5
- Training: 8
- Cool-down: 5

e#	Initial stored exercise
L	Learning action
L#	Learnt exercise reused

Experiments

- Experiment using JSHOP2 to test the plan generation
  - Average value of **intensity** and **difficulty** for 30 generated sessions



Discussion

- **Classical Planning**
  - The search is driven using action costs
  - Hard constraints to control the variability
  - Learning actions make backtracking among sessions unnecessary
- **HTN Planning**
  - Heuristic function handles the variability constraints and improves the planning time
  - It can plan multiple sessions without losing the possibility of backtracking
  - Axioms improve the expressiveness of the possible configurations of the model

Future Work

- Working on a better quantitative comparison
- Planners with better heuristics for fluents could improve planning time without the need of D&C strategy
- Temporal representation of the problem
- Implement *replanning* methods in case of updates after patient evaluation.

Clinical Decision Support System
(CDDSS)

Img. source: dryicons

THERAPIST

www.therapist.uma.es

Universidad Carlos III de Madrid

[plg]

Thank you for your attention

Universidad Carlos III de Madrid

Goal-directed Generation of Exercise Sets for Upper-Limb Rehabilitation

José C. Pulido, José C. González,

Arturo González-Ferrer, Javier García, Fernando Fernández,

Antonio Bandera, Pablo Bustos and Cristina Suárez

WS 5: Knowledge Engineering for Planning and Scheduling

See you in the poster session! ☺

Related work (Extra)

Clinical Decision Support Systems (CDSS) implement models based on expert knowledge to ease many tasks of physicians.

E.g.: Developing clinical practice guidelines.

- Mixed Integer Linear Programming (MILP) to determine appointments for patients of rehab hospitals.

  K. Schimmelpfeng et al.: "Decision support for rehabilitation hospital scheduling", 2012.

- Automated Planning for generating scenarios helping handicapped people.

  P. Morignot et al: "Generating scenarios for a mobile robot with an arm: Case study: Assistance for handicapped persons", 2010.

- Planning algorithm able to generate oncology treatment plans, including temporal constraints difficult to manage by physicians.

  A. González-Ferrer et al.: "Automated generation of patient tailored electronic care pathways by translating computer interpretable guidelines into hierarchical task networks", 2013.

STRIPS planning (Extra)

Metric-FF

```
 0: SESSION-START
 1: WARMUP-PHASE
 2: LEARN-WARMUP-EXERCISE
    SPATIAL_HAND LONG MAXIMUM
 3: TRAINING-PHASE
 4: TRAINING-DATABASE-EXERCISE E7
 5: LEARN-TRAINING-EXERCISE
          UNIMANUAL_COARSE
          MEDIUM MAXIMUM
 6: LEARN-TRAINING-EXERCISE
          UNIMANUAL_COARSE
          MEDIUM MAXIMUM
 7: TRAINING-DATABASE-EXERCISE E12
 8: LEARN-TRAINING-EXERCISE
          UNIMANUAL_FINE
          MEDIUM MAXIMUM
 9: COOLDOWN-PHASE
10: LEARN-COOLDOWN-EXERCISE
          SPATIAL_HAND LONG INTENSE
11: SESSION-END
```

CBP IPC-2011

```
 0: (SESSION-START)
 1: (WARMUP-PHASE)
 2: (WARMUP-DATABASE-EXERCISE E0)
 3: (TRAINING-PHASE)
 4: (TRAINING-DATABASE-EXERCISE E11)
 5: (TRAINING-DATABASE-EXERCISE E12)
 6: (TRAINING-DATABASE-EXERCISE E10)
 7: (TRAINING-DATABASE-EXERCISE E9)
 8: (LEARN-TRAINING-EXERCISE
          SPATIAL_HAND
          LONG RELAXED)
 9: (COOLDOWN-PHASE)
10: (COOLDOWN-DATABASE-EXERCISE E15)
11: (SESSION-END)
```

STRIPS Planning (Extra)

- Action to plan the learning of a new exercise

```
(:action learn-warmup-exercise
  :parameters (?t - target ?a - l_adequacy ?d - l_duration
    ?i - l_intensity)
  :precondition (and
    (current_phase p_warmup)
    (< (p_duration p_warmup) (p_max_duration p_warmup)))
  :effect (and
    (increase (exercise_index) 1)
    (increase (intensity_cumulative) (l_intensity_value ?a))
    (increase (p_duration p_warmup) (l_duration_value ?d))
    (increase (adequacy_cumulative ?t) (l_adequacy_value ?a))
    (increase (total-cost)
      (- (+ (l_adequacy_value ?a) 10) (l_duration_value ?d)))))
```

HTN Planning (Extra)

```
(:- (warmup-time ?current ?min ?max)
        ((call >= ?current 0) (wup-limit ?lw)
        (call <= ?current (call * (call / (call + ?min ?max) 2) ?lw)))
)
(:- (warmup-exercise ?e ?min ?max)
    ((wup-ex-config ?maxDuration ?maxIntensity ?maxDifficulty) (e-duration ?e ?d)
    (call >= ?d 1) (call <= ?d (call / (call * (call / (call + ?min ?max) 2) ?maxDuration) 2))
    (e-intensity ?e ?i) (call <= ?i ?maxIntensity)
    (e-difficulty ?e ?dif) (call <= ?dif ?maxDifficulty))
)
```

HTN Planning (Extra)

- Variability constraints are handled with a *sortby* function
- One execution of 20 sessions using:
  - **Heuristic function** ≈ 5 min.
  - **Relaxed Round Robin policy** > 30 min.

Pros and Cons (Extra)

	STRIPS	HTN
Search strategy	• CBP planner task, limiting learnings with action costs.	• Heuristic function to guide the search among the exercises.
Multiple sessions	• Divide-and-conquer strategy. • Impedes backtracking among sessions, but it is not needed.	• Can plan as usual, in one run.
Avoids repeated exercises	• In the last 3 sessions. • In the same position than in the last occurrence.	• In the same session. • Penalizes repeated exercises, but allows them.
Learning	• Suggests attribute values. • Prefers exercises which improve variability.	• Adds new predefined exercises during planning time.
Phase parameterization	• Controlled by predicates and functions.	• Axioms allows to model expert knowledge easily.