Sistemas Multi-Agentes

Aprendizaje Intencional

Dr. Alejandro Guerra-Hernández

Universidad Veracruzana
Instituto de Investigaciones en Inteligencia Artificial
Campus Sur, Calle Paseo Lote II, Sección Segunda No 112,
Nuevo Xalapa, Xalapa, Ver., México 91097

aguerra@uv.mx
https://www.uv.mx/personal/aguerra/

Maestría en Inteligencia Artificial 2023

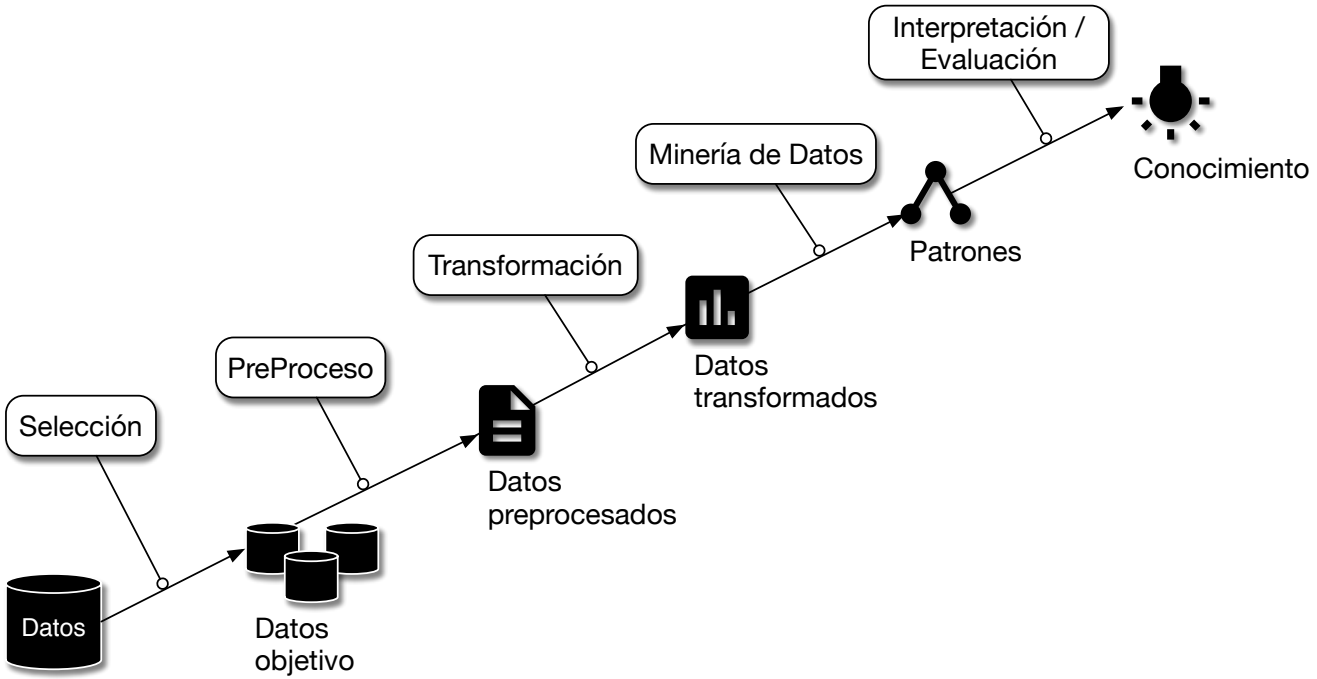
Aprendizaje

- Se dice que un programa de computadora, agentes incluídos, aprende de una experiencia E, con respecto a una clase de tareas T y una medida de desempeño P, si su desempeño en las tareas en T, tal y como es medido por P, mejora con la experiencia E –Mitchell [4].
- Ejemplo: Damas chinas.
  - Tareas T: Jugar damas chinas.
  - Medida de desempeño P: Porcentaje de partidas perdidas.
  - Experiencia de entrenamiento E: Jugar contra si mismo (Aprendizaje por refuerzo, por ejemplo Q-Learning).
- Aprendizaje $\neq$ Minería de Datos

Universidad Veracruzana

Minería de datos

- Fayyad, Piatetsky-Shapiro y Smyth [1] definen el proceso de extracción de conocimiento a partir de datos, como sigue:



- La experiencia E ha sido substituida por los datos.

Universidad Veracruzana

Problemas de diseño en aprendizaje

- El tipo exacto de conocimiento a ser aprendido.
  - Una calificación del tablero resultante $V(t) \mapsto \mathbb{R}$:
    - Si gano $V(t) = 100$;
    - Si pierdo $V(t) = -100$;
    - Si empato $V(t) = 0$.
    - Para tableros no finales $V(t) = V(t')$, donde t' es el mejor tablero posible a partir de t, al estilo Q-Learning.
- La representación de ese conocimiento objetivo.
  - Propiedades ponderadas del tablero: x_1 piezas negras, x_2 piezas rojas, x_3 negras coronadas, x_4 rojas coronadas, x_5 negras amenazadas, x_6 rojas amenazadas:

$$\hat{V}(t) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 + w_5 x_5 + w_6 x_6$$

- El mecanismo de aprendizaje.
  - LMS (*Least mean squares*)

Universidad Veracruzana

LMS

- Dado el error cuadrático entre los valores de entrenamiento y los valores predichos por la hipótesis:

$$Error \equiv \sum_{\langle t, V_{train}(t) \rangle \in Exs} (V_{train}(t) - \hat{V}(t))^2$$

- Buscar los pesos $w_{i=1,\ldots,6}$ $(\hat{V})$ que minimizan el error cuadrático, usando la regla de actualización LMS:
  - Por cada ejemplo de entrenamiento $\langle t, V_{train}(t) \rangle$
    - Use los pesos actuales para computar $\hat{V}(t)$.
    - Para cada peso w_i actualizelo como sigue:

$$w_i \leftarrow w_i + \eta(V_{train}(t) - \hat{V}(t))\ x_i$$

- ¿Porqué funciona la regla?

Universidad Veracruzana

Módulos de un agente aprendiz

Actuador. Es el módulo que usa la función aprendida para llevar a cabo las tareas en T.

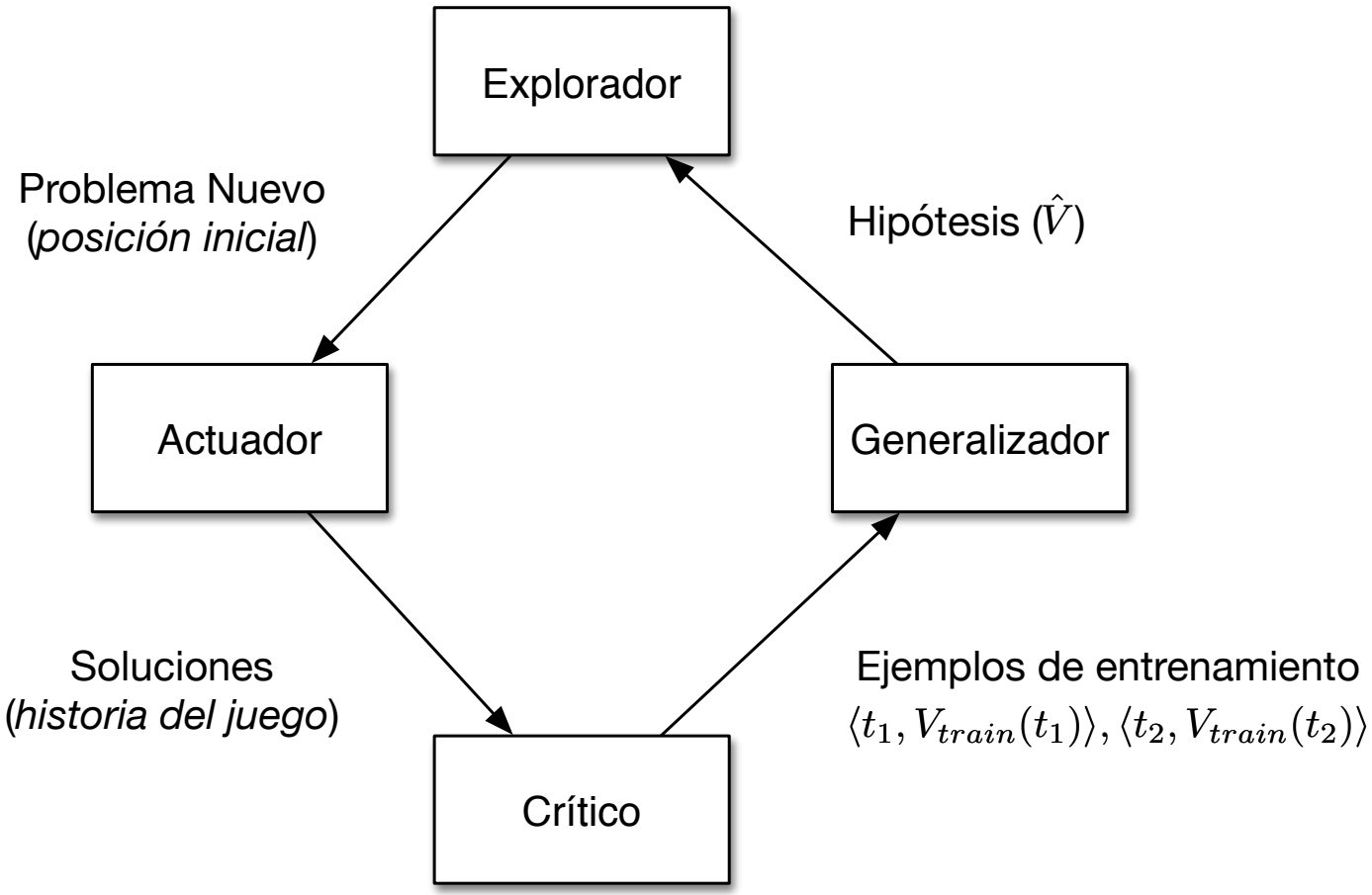
Crítico. Es el módulo que produce ejemplos de entrenamiento a partir de la experiencia E.

Generalizador. El módulo que produce una hipótesis que es un estimado de la función objetivo. El algoritmo de aprendizaje.

Explorador. El módulo encargado de proponer nuevos problemas para explorar la hipótesis aprendida hasta el momento.

Universidad Veracruzana

Interacción entre módulos



Universidad Veracruzana

Consideraciones (Russell y Norvig [5])

- ¿Qué componente del agente será mejorado?
- ¿Qué conocimiento previo hay disponible?
- ¿Qué representación es usada para los datos y componentes?
- ¿Qué retroalimentación se usa para aprender?

Universidad Veracruzana

Componentes

- Un mapeo directo enttre el estado del ambiente y las acciones.
- Un medio de inferencia de las propiedades del ambiente a partir de las secuencias de percepción.
- Información sobre cómo evoluciona el ambiente y sobre las acciones que se pueden tomar.
- Utilidades indicando que tan deseables son los estados.
- Valores de acción indicando que tan deseables son éstas.
- Metas que describen los estados que maximizan la utilidad obtenida.

Universidad Veracruzana

Componentes *AgentSpeak(L)*

- El contexto de un plan indica cuando es racional formar una intención con éste: Cuando es relevante y aplicable.
- La aplicabilidad de un plan expresa las condiciones bajo las cuales se puede formar una intención con el plan, esperando que ésta tenga éxito.
- Esta información de contexto se puede aprender, invividual y socialmente, por ejemplo en la imitación contextual (Hoppitt y Laland [3]).

Universidad Veracruzana

Representaciones *AgentSpeak*(*L*)

- A diferencia del aprendizaje tradicional, donde las representaciones son proposicionales; las representaciones *AgentSpeak*(*L*) son de primer orden.
- El conocimiento previo de un agente *AgentSpeak*(*L*) es basto, se puede aprender tomando en cuenta lo que se sabe del problema (Las creencias de un agente) y lo que se sabe de las habilidades del agente (Sus planes).

Universidad Veracruzana

Retroalimentación en *AgentSpeak(L)*

- Tradicionalmente, los métodos de aprendizaje utilizan una retroalimentación:

  Supervisada. Los ejemplo están etiquetados con su clase.
  No supervisada. Los ejemplos no lo están.
  Por refuerzo. Los estados del ambiente o las acciones son evaluados con una calificación.

- La señal inmediatamente disponible para aprender es el éxito (o el fallo) de las intenciones, que puede representarse de manera supervisada o por refuerzo.

Universidad Veracruzana

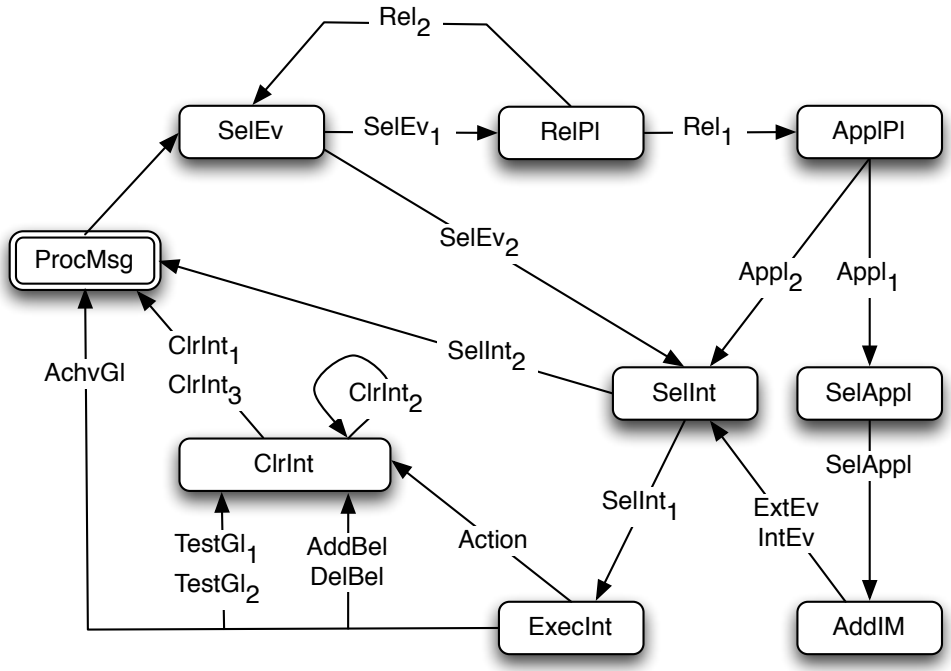
JILDT

- Un modelo de aprendizaje BDI bajo el razonamiento práctico *à la* Bratman (Guerra-Hernández, El-Fallah-Seghrouchni y Soldano [2]).
- Las razones prácticas para adoptar un plan como intención están en el contexto de los planes.
- Los ejemplos de entrenamiento a partir de estados mentales del agente: Creencias, Intención actual, etiqueta de éxito o fracaso.
- Se aprende cuando adoptar una intención y también cuando abandonarla, como en el compromiso basado en políticas.

$$JILDT = \text{Jason} + \text{Induction of Logical Decision Trees}$$

Universidad Veracruzana

Justificación




```
1   // Beliefs
2   on(b,a).
3   on(a,table).
4   on(c,table).
5   on(z,table).
6
7   // Rules Believed
8   clear(X) :- not(on(_,X)).
9   clear(table).
10
11  // Desires
12  !put(b,c).
13
14  // Plans (bold agent)
15  @put
16  +!put(X,Y) : true
17    <- move(X,Y).
```

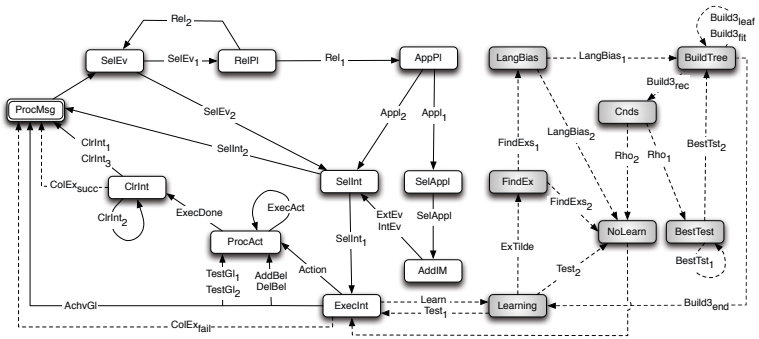
- ¿Qué sucede si `move` falla?
- La intención falla aún cuando las razones prácticas eran correctas!

Universidad Veracruzana

Top-Down Induction of Logical Decision Trees (TILDE)

Base conocimientos (Ejemplos)

begin(model(1))
fail.
intend(put,b,c).
on(b,a).
on(a,table).
on(c,table).
on(z,c).
end(model(1))



Bias lenguaje (Candidatos)

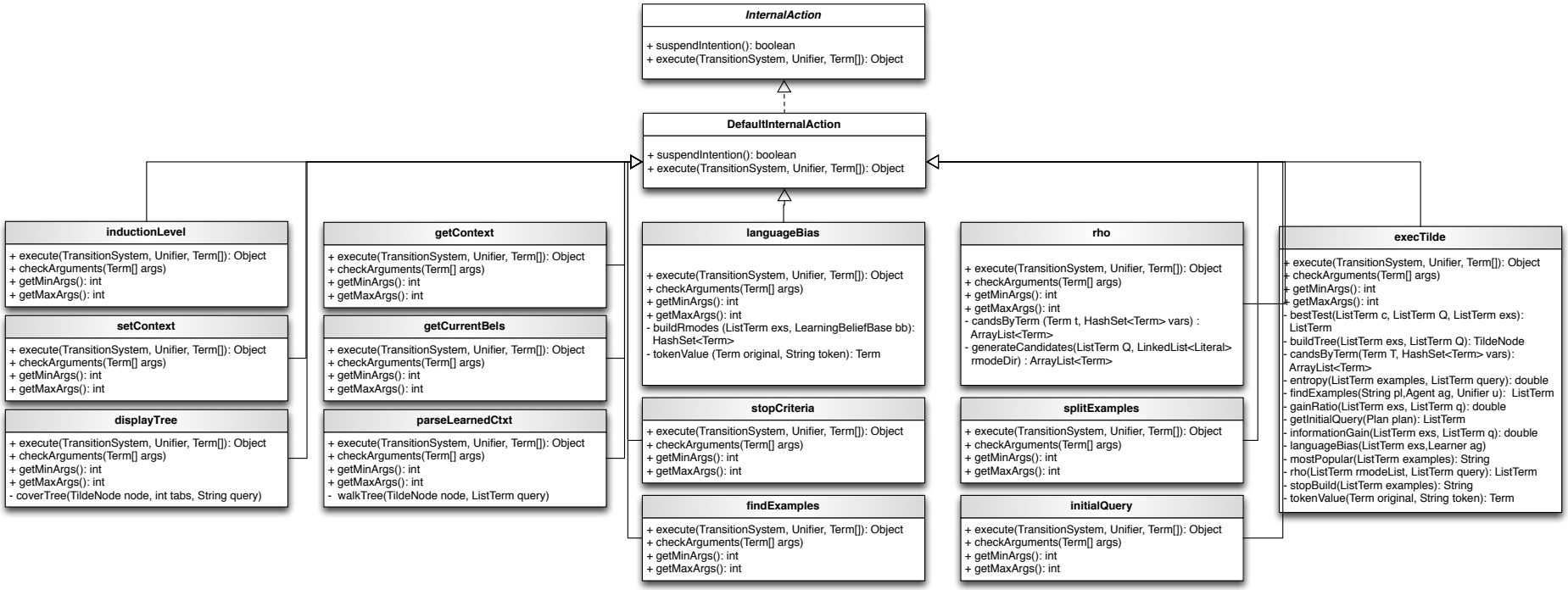
rmode(clear(+V1)).
rmode(on(+V1,+V2)).
...
lookahead(intend(put,V1,V2),clear(V1)).
lookahead(intend(put,V1,V2),clear(V2)).
...

Conocimiento previo (Reglas)

clear(X) :- not(on(_,X)).
clear(table).

Universidad Veracruzana

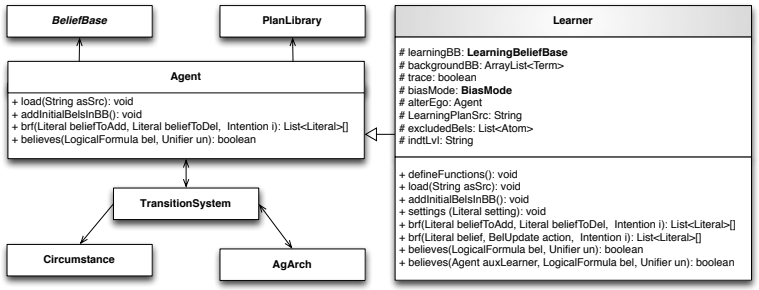
Acciones internas: jildt.*action*



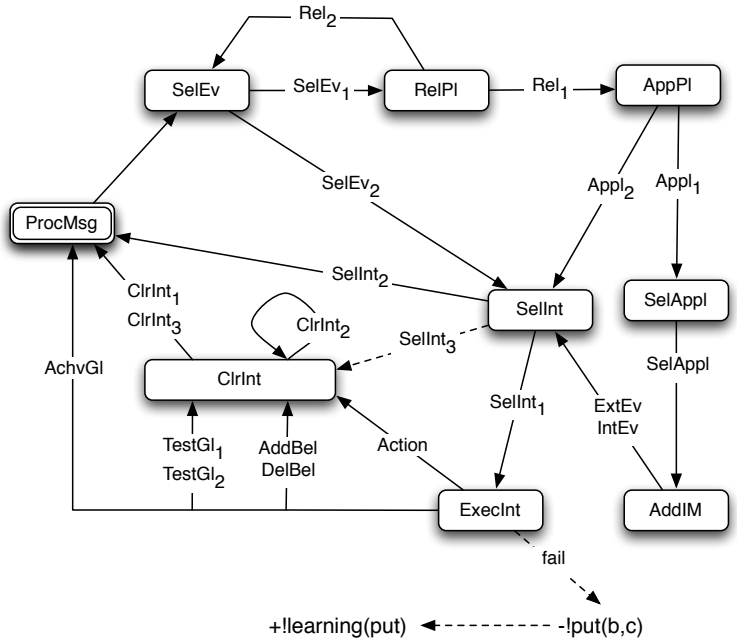
Universidad Veracruzana

Clases de Agente

- IntentionalLearner



- SingleMindedLearner



Planes aprendizaje / abandono intención

```
1  @learning
2  +!learning(P): true <-
3        .print("Trying to learn a better context...");
4        jildt.setTilde(P);
5        jildt.execTilde(false,false);
6        jildt.getLearnedCtxt(P,LC,F);
7        !learningTest(P,LC,F).
8
9  @dropIntention
10 +dropIntention(I) : true <-
11     .print("Sorry, dropping my intention");
12     -intending(I,_);
13     .drop_intention(I);
14     +dropped_int(I).
```

Universidad Veracruzana

Evaluando planes aprendidos

```
1  @learningTestSuccess
2  +!learningTest(P,LC,change): true <-
3       .print("Learned context for ",P," is ", LC);
4       .print("Changing context in plan " , P, " ...");
5       jildt.changeCtxt(P,LC);
6       .print("Context was changed successfully...");
7       jildt.addDropRule(P,LC).
8
9  @learningTestFail
10 +!learningTest(P,LC,notchange) : true <-
11      .print("It's not possible to learn better context").
```

Universidad Veracruzana

Plan extendido

```
1  @[put]
2  +!put(X,Y) : true <-
3       jildt.getCurrentInt(I);
4       jildt.getCurrentBels(Bs);
5       +intending(I,Bs);
6       move(X,Y);
7       -intending(I,Bs);
8       +example(I,Bs,succ).
```

Universidad Veracruzana

Fallo por ausencia de plan relevante

```
1  @put_failCase
2  -!put(X,Y) : intending(put(X,Y), Bs) <-
3       -intending(I,Bs);
4       +example(I,Bs,fail);
5       !learning(put);
6       +example_processed(put).
7
8  @put_failCase_NoRelevant
9  -!put(X,Y) : not intending(put(X,Y),_) <-
10      .print("Plan ",put," non applicable.");
11      +non_applicable(put).
```

Universidad Veracruzana

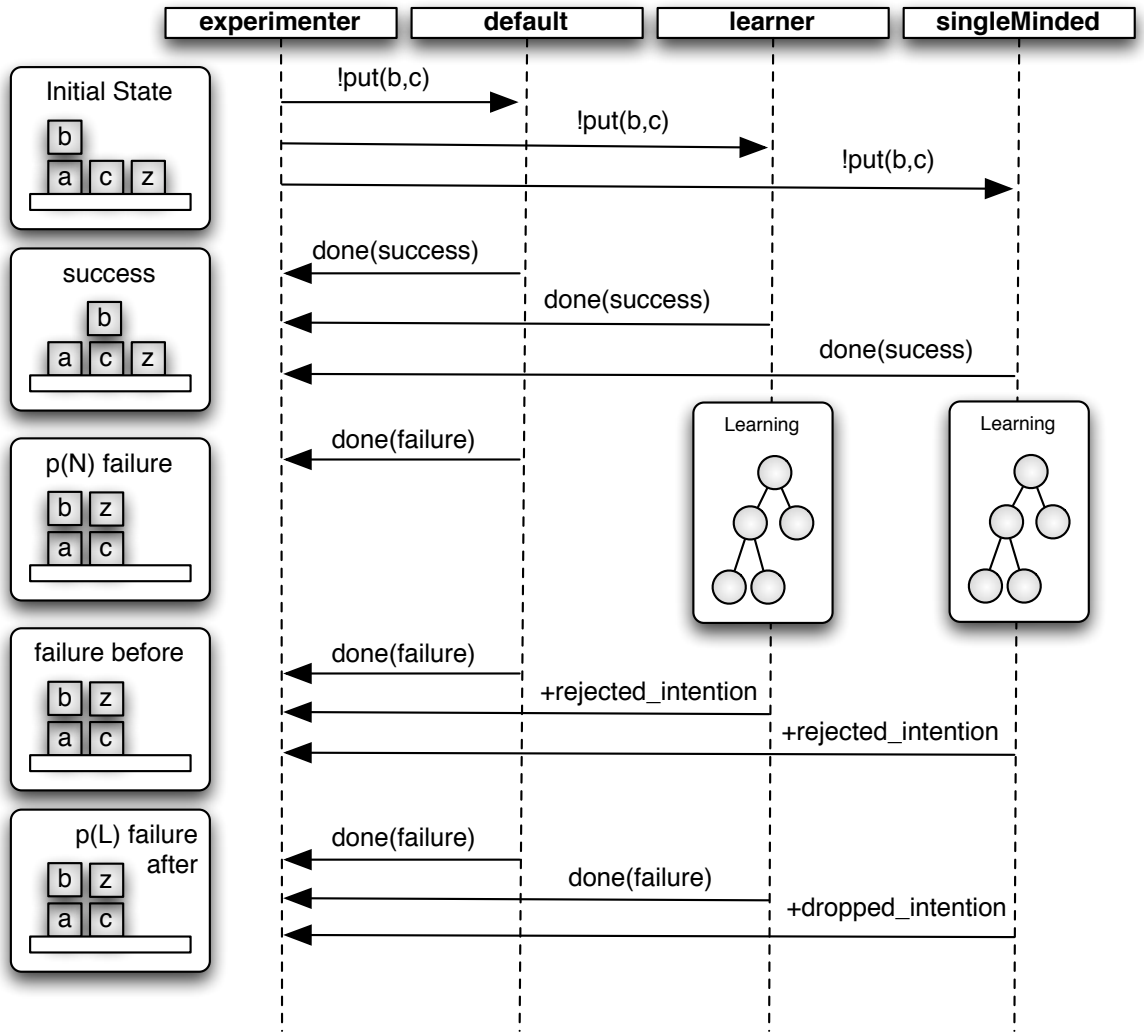
Algoritmo de aprendizaje (jildt.tilde)

1: **procedure** buildTree(E,Q)
2: $\leftarrow Q_b := best(\rho(\leftarrow Q))$ ▷ E is a set of examples, Q a query
3: **if** $stopCriteria(\leftarrow Q_b)$ **then** ▷ *best* max information gain
4: $T := leaf(majority\_class(E))$ ▷ E.g., No information gain obtained
5: **else**
6: $Conj \leftarrow Q_b \backslash Q$
7: $E_1 \leftarrow \{e \in E \mid e \wedge B \models Q_b\}$
8: $E_2 \leftarrow \{e \in E \mid e \wedge B \not\models Q_b\}$
9: $buildTree(Left, E_1, Q_b)$;
10: $buildTree(Right, E_2, Q)$
11: $T \leftarrow nodei(Conj, Left, Right)$
12: **end if**
13: **return** T ▷ The built tree
14: **end procedure**

- Implementado como una acción interna

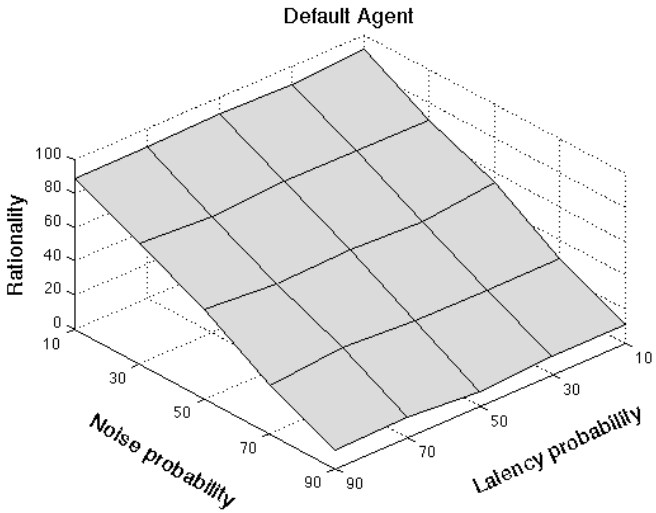
Universidad Veracruzana

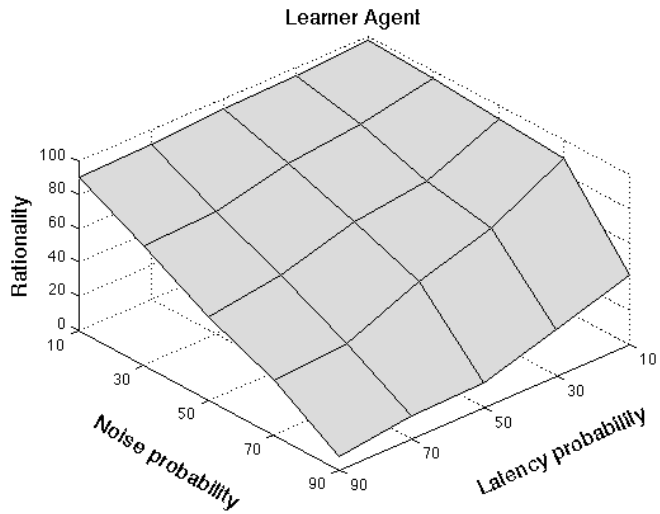
Experimento

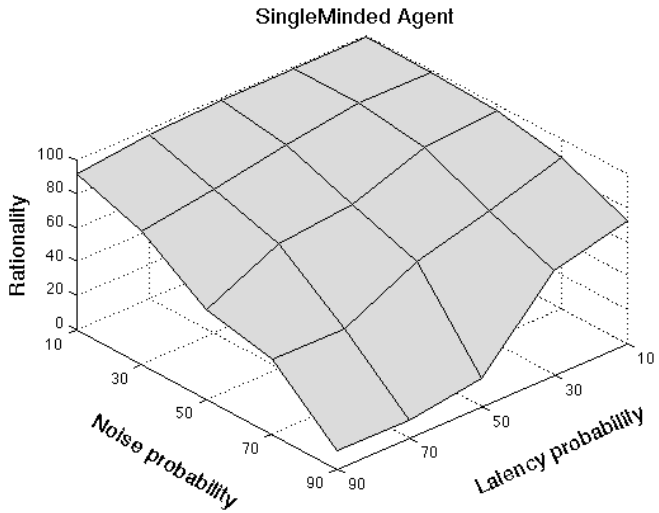


Universidad Veracruzana

Racionalidad

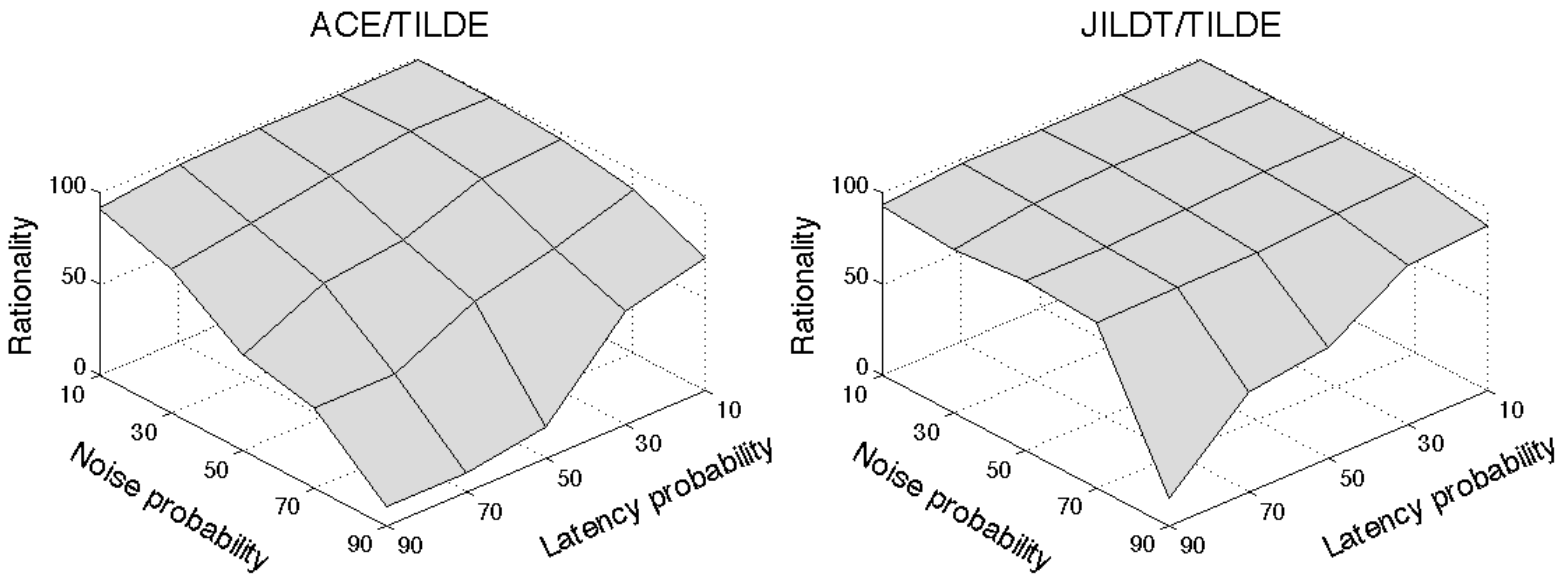






Universidad Veracruzana

Comparativa ACE vs JILDT (single minded)



ersidad Veracruzana

Comentarios

- Jildt no está pensado para llevar a cabo Minería de datos, sino aprendizaje intencional: Una revisión de la formación y abandono de intenciones vía aprendizaje.
- Para hacer Minería de Datos habría que usar artefactos basados en Tilde.
- El aprendizaje intencional provee el grado adecuado de confianza-cautela con respecto al compromiso.
- Las diferencias ACE vs Jildt pueden deberse a que Jildt no poda.

Universidad Veracruzana

Descarga de Jildt

- Repositirio: `http://sourceforge.net/projects/jildt/files/`
- Página web: `http://jildt.sourceforge.net`
- Requisitos:
  - Jason >= 1.3.X
  - Java >= 1.6

Universidad Veracruzana

Referencias I

[1] U Fayyad, G Piatetsky-Shapiro y P Smyth. "From Data Mining to Knowledge Discovery in Databases". En: *AI Magazine* 17.3 (1996), págs. 37-54.

[2] A Guerra-Hernández, A El-Fallah-Seghrouchni y H Soldano. "Learning in BDI Multi-agent Systems". En: *Computational Logic in Multi-Agent Systems: 4th International Workshop, CLIMA IV, Fort Lauderdale, FL, USA, January 6–7, 2004, Revised and Selected Papers*. Ed. por J Dix y J Leite. Vol. 3259. Lecture Notes in Computer Science. Berlin, Germany: Springer-Verlag, 2004. Cap. Learning in BDI Multi-agent Systems, págs. 218-233.

[3] W Hoppitt y KN Laland. *Social Learning: An Introduction to Mechanisms, Methods, and Models*. Princeton, NJ, USA: Princeton University Press, 2013.

[4] T Mitchell. *Machine Learning*. Computer Science Series. Singapore: McGraw-Hill International Editions, 1997.

[5] SJ Russell y P Norvig. *Artificial Intelligence: A Modern Approach*. Third. Prentice Hall Series in Artificial Intelligence. USA: Prentice Hall, 2009.

Universidad Veracruzana