



# Adaptive strategies for mining the positive border of sentences

---

Jean-Marc Petit

LIMOS, UMR 6158 CNRS  
Clermont-Ferrand, FRANCE

Joint work with

Fabien De Marchi, Frederic Flouvat

# Outline

---

- **Notations**
- Adaptive algorithms
  - Links between borders
  - Principles of our contribution
- Applications
  - Inclusion dependencies
  - Maximal frequent itemsets (ongoing)
- Related contributions
- Conclusion

# Notations

---

- A model for KDD [Mannila & Toivonen, 1997]:
  - A database  $d$
  - A finite language  $L$
  - A partial order  $\leq$  on sentences of  $L$
  - An anti-monotonic predicate  $Q$ 
    - $Q(X,d)$  true iff  $X$  is “interesting” in  $d$  wrt  $Q$
    - $\forall X,Y \in L$  such that  $X \leq Y$ ,  $Q(Y,d)$  true  $\Rightarrow Q(X,d)$  true
  - $L$  is « representable » as a set  $E$ 
    - $(L, \leq)$  isomorphic to  $(E, \subseteq)$ , i.e. there exists a bijective function  $f: L \rightarrow 2^E$  such that  $X \leq Y \Leftrightarrow f(X) \subseteq f(Y)$

# Notations (cont'd)

---

- Let **I** the interesting subsets of E
- The **positive border** of I (or the *most specific sentences* of I), noted **Bd<sup>+</sup>(I)**, is:

$$Bd^+(I) = \{X \in I \mid \forall Y \supset X, Y \notin I\}$$

- The **negative border** of I, noted **Bd<sup>-</sup>(I)**, is made up of « not interesting subsets » of E :

$$Bd^-(I) = \{X \in (2^E \setminus I) \mid \forall Y \subset X, Y \in I\}$$

- Problem statement:

Enumerate elements of **Bd<sup>+</sup>(I)**

# Outline

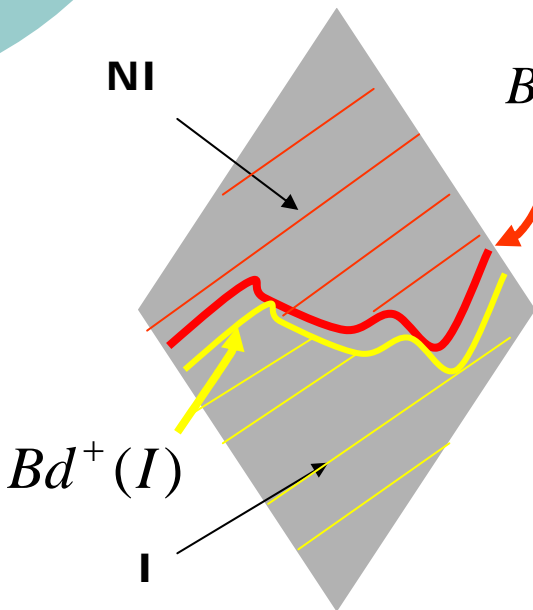
---

- Notations
- Adaptive algorithms
  - Links between borders
  - Principles of our contribution
- Applications
  - Inclusion dependencies
  - Maximal frequent itemsets (ongoing)
- Related contributions
- Conclusion

# Links between borders

- Minimal transversals of **hypergraph** [Mannila & Toivonen 97]
- Takes its roots from Functional Dependency inference
  - [Mannila & Raihä 94, Demetrovics & Thi 95]

$I = \{X \subseteq E \mid X \text{ interesting}\}$     $NI = \{X \subseteq E \mid X \text{ not interesting}\}$



$$\begin{aligned}
 Bd^-(I) &= \min \{X \subseteq E \mid X \in NI\} \\
 &= \min_{\subseteq} \{X \subseteq E \mid \forall Y \in Bd^+(I), X \not\subseteq Y\} \\
 &= \min_{\subseteq} \{X \subseteq E \mid \forall Y \in Bd^+(I), X \cap (E \setminus Y) \neq \emptyset\}
 \end{aligned}$$

Let  $\overline{Bd^+(I)} = \{X \subseteq E \mid E \setminus X \in Bd^+(I)\}$  an hypergraph

$$Bd^-(I) = \text{MinTr}(\overline{Bd^+(I)})$$

**Mintr(H)** : minimal transversals of H

# Links between borders (cont'd)

---

- Property [Berge74]:

Let  $H$  be an hypergraph.  $\text{MinTr}(\text{MinTr}(H)) = H$

- From the previous result, we get :

$$Bd^+(I) = \overline{\text{MinTr}(Bd^-(I))}$$

- Rather surprisingly, less studied in that direction
- Might be useful if  $Bd^-(I)$  is known

# Principles of our contribution

---

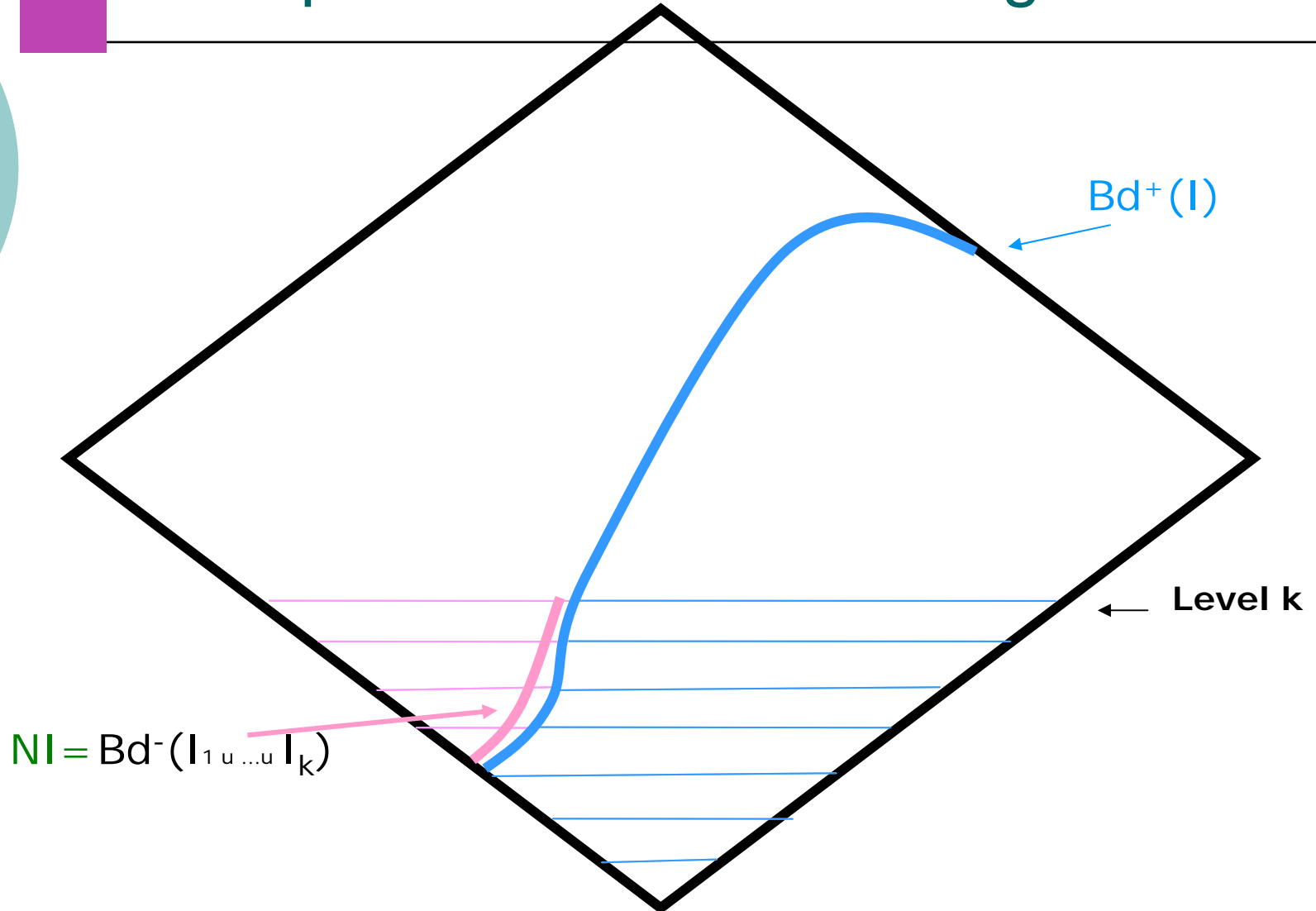
- **Step 1:** Guess  $NI$ , a subset of  $Bd^-(I)$
- **Step 2:** Compute  $Bd^+_{opt}(I)$  from  $NI$ , i.e. the *optimistic positive border*
- **Step 3:** Estimate of the **quality** of  $Bd^+_{opt}(I)$  wrt  $Bd^+(I)$  (to be discovered)
- **Step 4:** Based on the estimates, **guide** the remaining traversal of the search space
  - Iteration to **step 2** might be possible

## Step 1: Computing a subset of $Bd^-(I)$

---

- Many available strategies such as level-wise algorithms
  - Up to a given level, say  $k$
  - *Well-adapted whenever large elements in  $Bd^+(I)$  exist*
- The “best strategy” should avoid interesting sets !
  - Application-dependent
  - Data-dependent

# Example with a level-wise algorithm



## Step 2: the optimistic positive border

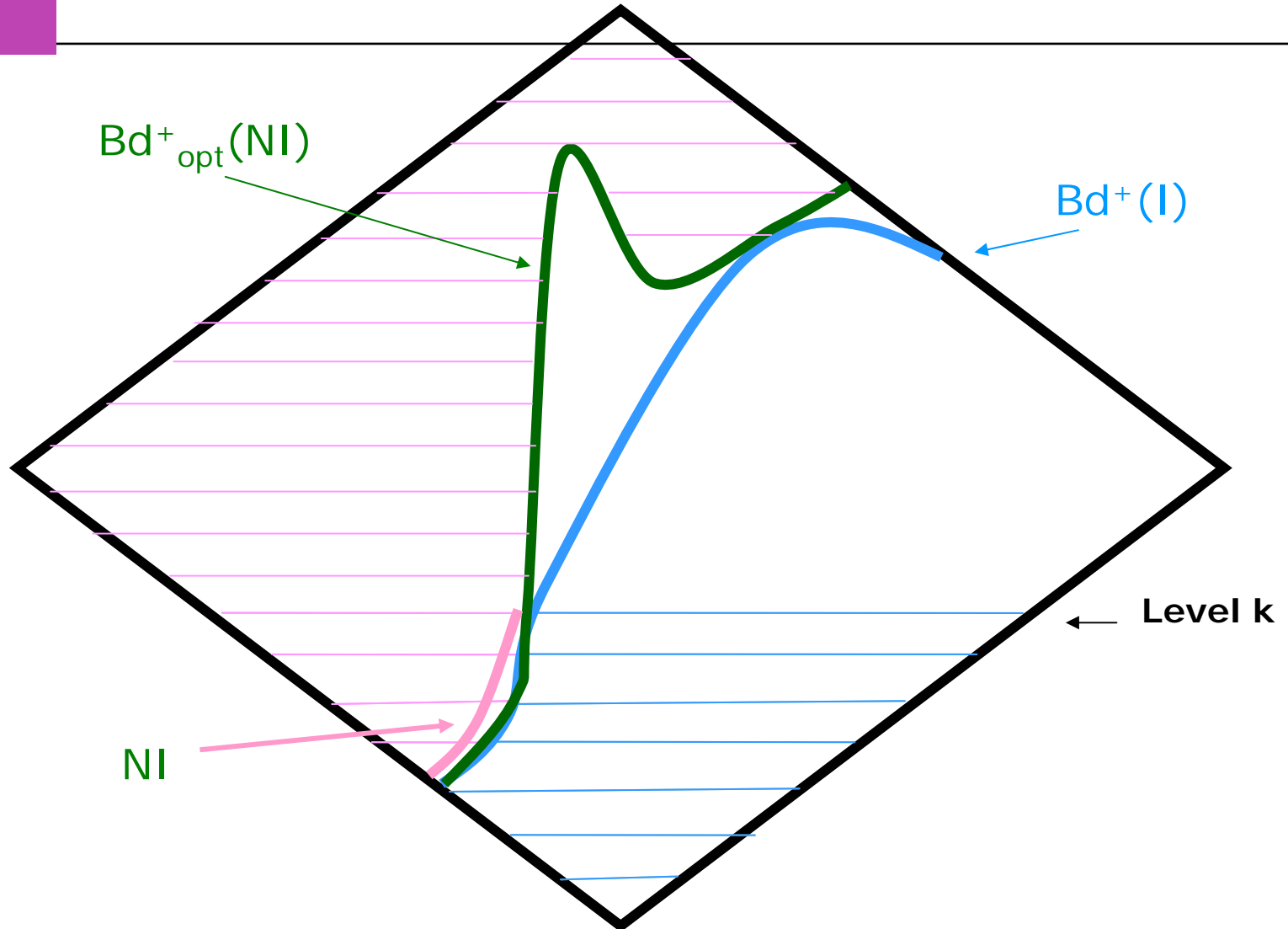
---

- The *optimistic positive border* is the set of greatest elements not yet disqualified by NI
- From previous slides, we have :

$$Bd_{opt}^+(\mathbf{I}) = \overline{\text{MinTr}(NI)}$$

- Tightly coupled with the notion of “jump” in the search space

# Intuition

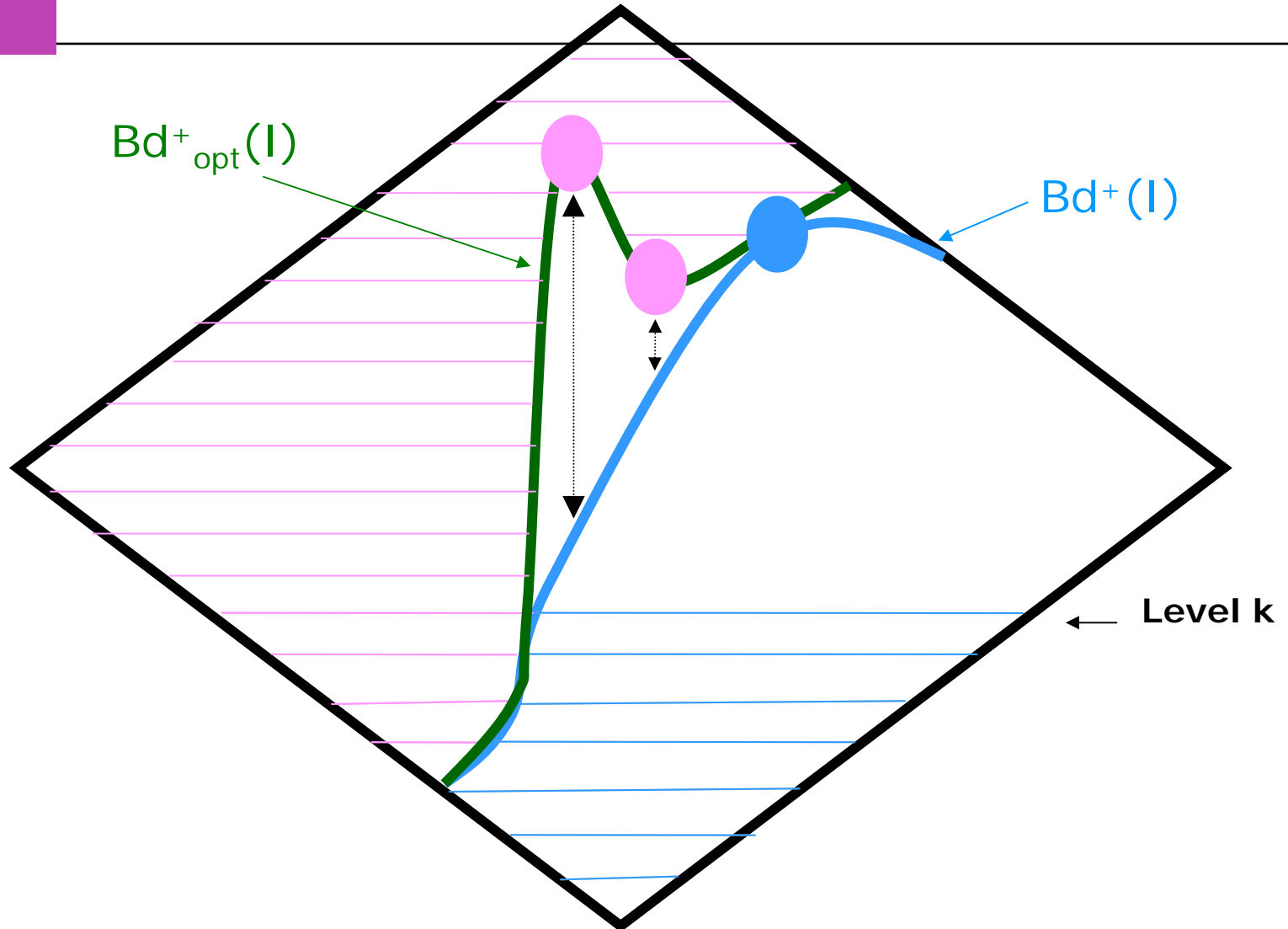


# Step 3: Estimate of the error

---

- How to evaluate this optimistic positive border ?
- Let  $X \in \text{Bd}^+_{\text{opt}}(I)$ 
  - Either  $Q(X,d)$  true 😊
  - Or  $Q(X,d)$  false 😞
    - $\text{Error}(X,d)$  should try to quantify the distance between  $X$  and  $Y$  with  $Y \subseteq X$  and  $Y \in \text{Bd}^+(I)$
    - Must be a *monotone function*
      - $X \subseteq Y \Rightarrow \text{error}(X,d) \leq \text{error}(Y,d)$
- Application-dependent

# Intuition

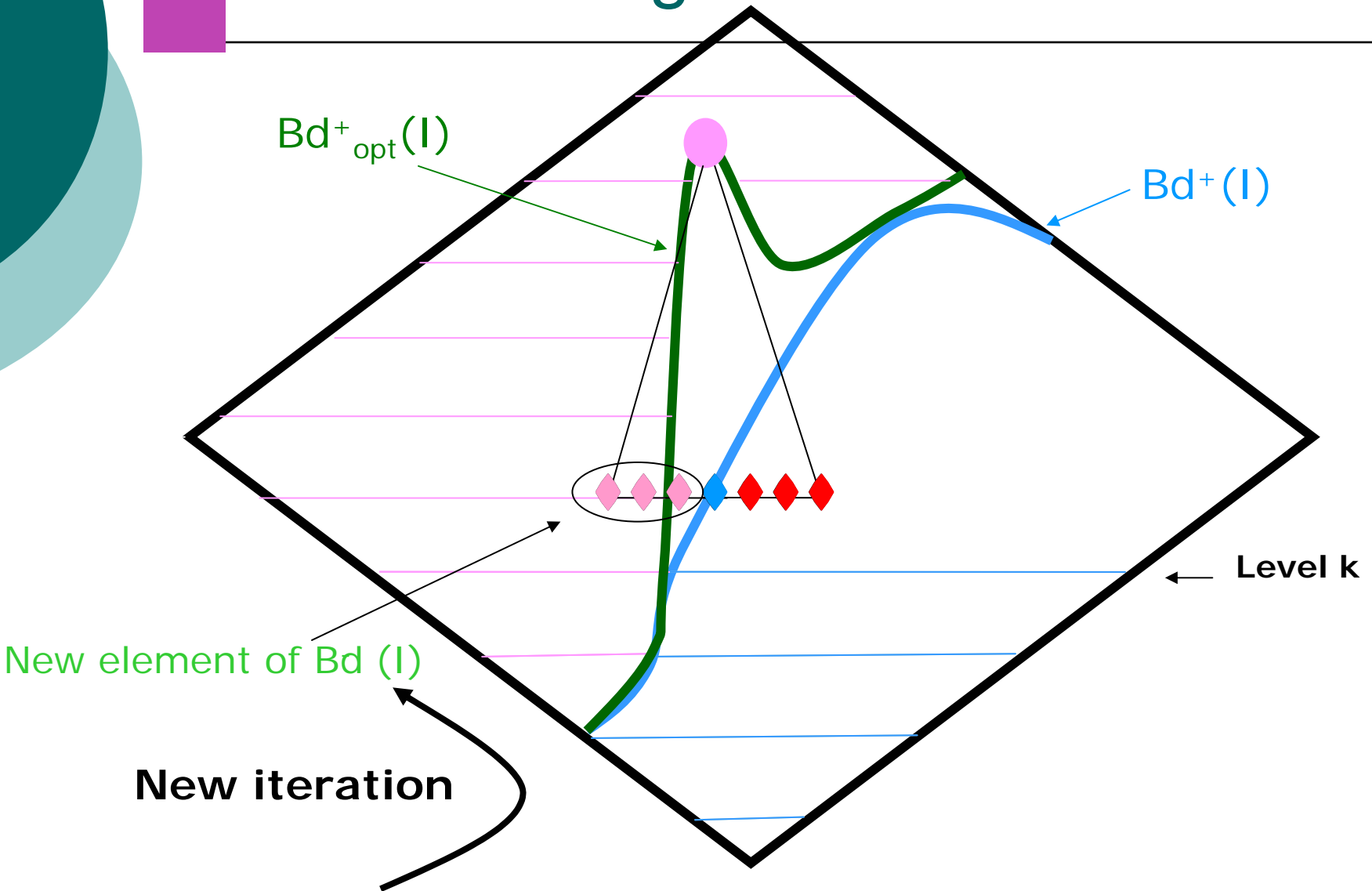


# Step 4: Adaptive strategy

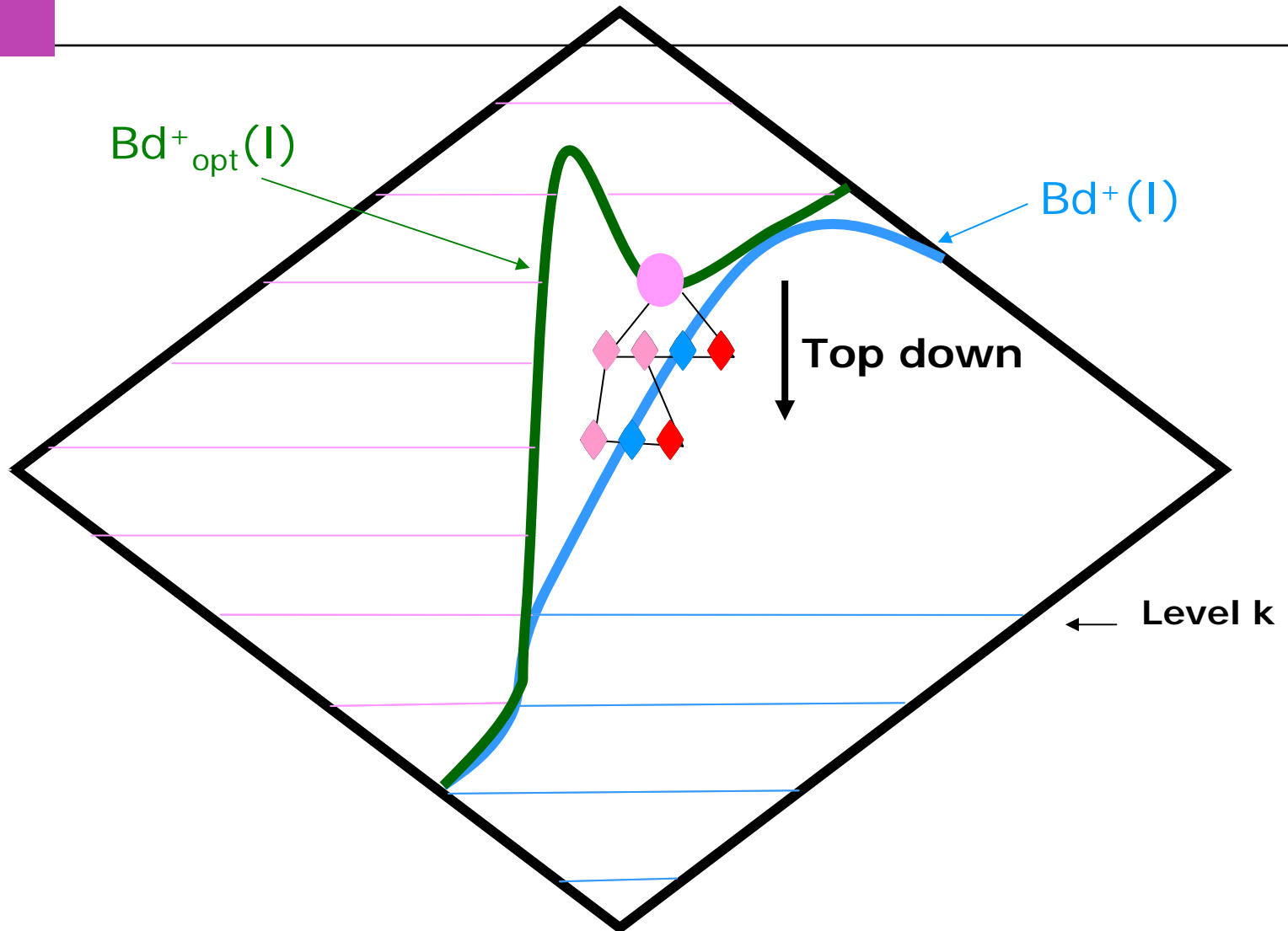
---

- Basic idea :
  - Guiding the search
  - Try to adapt the traversal of the search space wrt information discovered so far
    - Avoid to enumerate the largest antichains of  $(E, \subseteq)$
- Example of strategy:
  - Given a threshold  $\varepsilon$  and  $X \in \text{Bd}^+_{\text{opt}}(I)$
  - $X \in \text{error}(Q(X, d)) > \varepsilon$ 
    - The solution should be far away
      - “Bottom-up approach” for each  $X$  and iteration
  - $\text{error}(Q(X, d)) \leq \varepsilon$ 
    - The solution should not be too far
      - Top-down approach, no iteration
- Many other strategies may be devised

# Intuition : large error



# Intuition : small error



# Complexity issue

---

- Specific to
  - step 1: the initialization
  - step 4: the chosen strategy
- Simple result:
  - let  $k$  the largest set of  $Bd^-(I)$ . Assume a levelwise algorithm from level 1 to level  $k$  has been used. Then
    - The *number of queries* is bounded by  $O(2^k |Bd^-(I)| + |Bd^+(I)|)$
    - The *number of DB scan* is equal to  $k+1$

# Outline

---

- Notations
- Adaptive algorithms
  - Links between borders
  - Principles of our contribution
- Applications
  - Inclusion dependencies
  - Maximal frequent itemsets (ongoing)
- Related contributions
- Conclusion

# Application to inclusion dependencies

---

- Problem statement
  - « Given a database  $d$ , find the positive border of satisfied IND in  $d$  »
- IND defined on attribute *sequences*, *not on sets*
- That problem fits into the theoretical framework
  - Isomorphism with  $(E, \subseteq)$  not so easy
- ZigZag algorithm:
  - Step 1: levelwise algorithm until level  $k$  [EDBT'02]
  - Step 2: Adaptation of Demetrovics & Thi algorithm for minimal transversals computation
  - Step 3: error measure, number of values to be removed to get a satisfied IND
  - Step 4: as described earlier

# Experimental results (1)

---

$ \text{Bd}^+(\text{I}) $	Size of largest INDs	MIND (Levelwise)	Zigzag
5	6	446 s.	237 s.
10	7	2 790 s.	1 754 s.
10	11	25 626 s.	3 500 s.
20	18	> 1 year	7 729 s.

- $k = 2, \varepsilon = 1$
- The cost of one DB access is **very high**

# Experimental results (2)

---

- Remark:
  - $Bd^+_{opt}(I)$  almost equals to  $Bd^+(I)$  at the first iteration !
- Optimistic jumps justified by an interaction property between FD and IND [Mitchell83]
  - if  $R[UV] \subseteq S[XY]$   
and  $R[UW] \subseteq S[XZ]$   
and  $S:X \rightarrow Y$   
then  $R[UVW] \subseteq S[XYZ]$
- Cf details in [ICDM'03]

# Application to maximal frequent itemsets (ongoing)

---

- Challenging application
- Up to now, the choices made are:
  - Step 1: Apriori algorithm until level k
    - Value of k computed **dynamically**
      - When  $F_{i+1}/C_{i+1}$  exceeds a given threshold
  - Step 2: *as for IND*
  - Step 3:  $\text{Error}(X,d) = \text{minsup} - \text{frequency}(X,d)$
  - Step 4: *as given in the general framework*
- Implementation:
  - adaptation of Bart Goetals implantation of Apriori [*FIMI repository*]

# Maximal frequent itemsets (cont'd)

---

- First feedbacks:
  - Very sensitive to
    - error measure
      - Even for small errors, they may remain the same on several levels
    - The used strategies
  - Minimal transversals computation not prohibitive
- Experimental evaluations ongoing

# Outline

---

- Notations
- Adaptive algorithms
  - Links between borders
  - Principles of our contribution
- Applications
  - Inclusion dependencies
  - Maximal frequent itemsets (ongoing)
- **Related contributions**
- Conclusion

# Main contributions

---

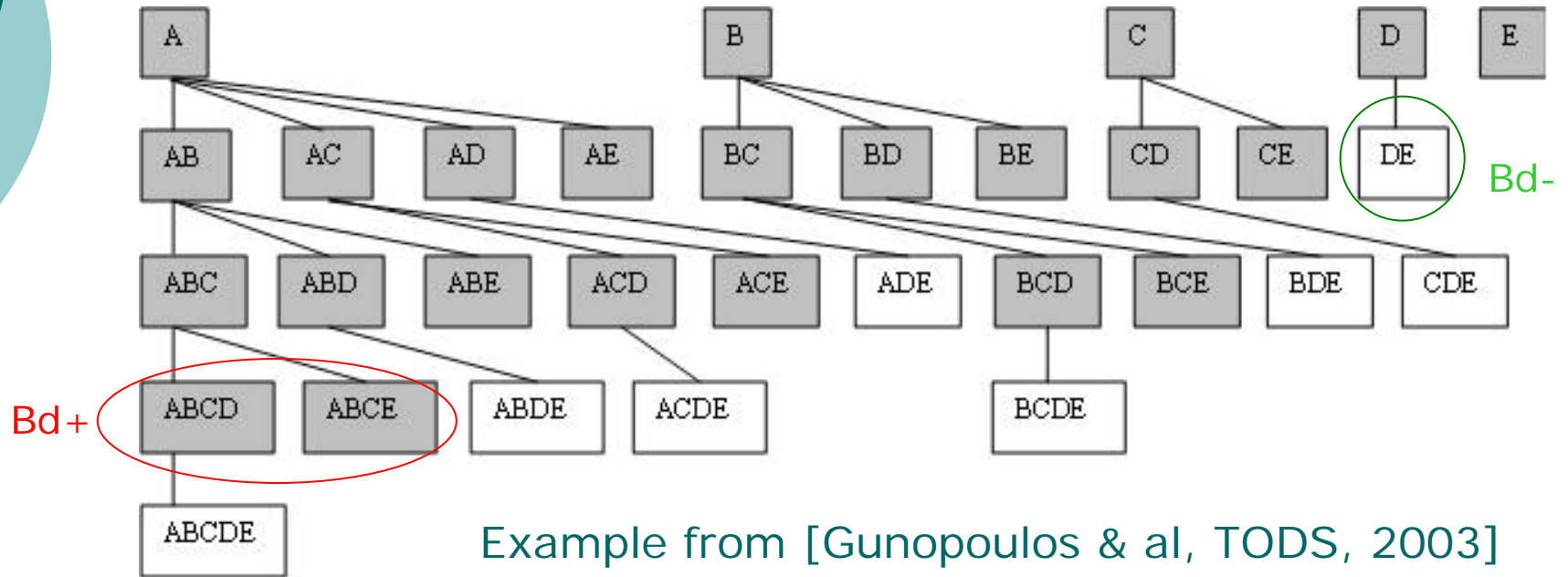
- Maximal frequent itemsets
  - Pincer Search [EDBT'98]
  - MaxMiner [Sigmod'98], GenMax [KDD'01], Mafia [ICDE'01], ...
- Inclusion dependencies
  - Köller [ICDE'03]
- Theoretical framework
  - Dualize and Advance [ICDT'97,PODS'97,TODS'03]

# Pincer Search [EDBT'98]

---

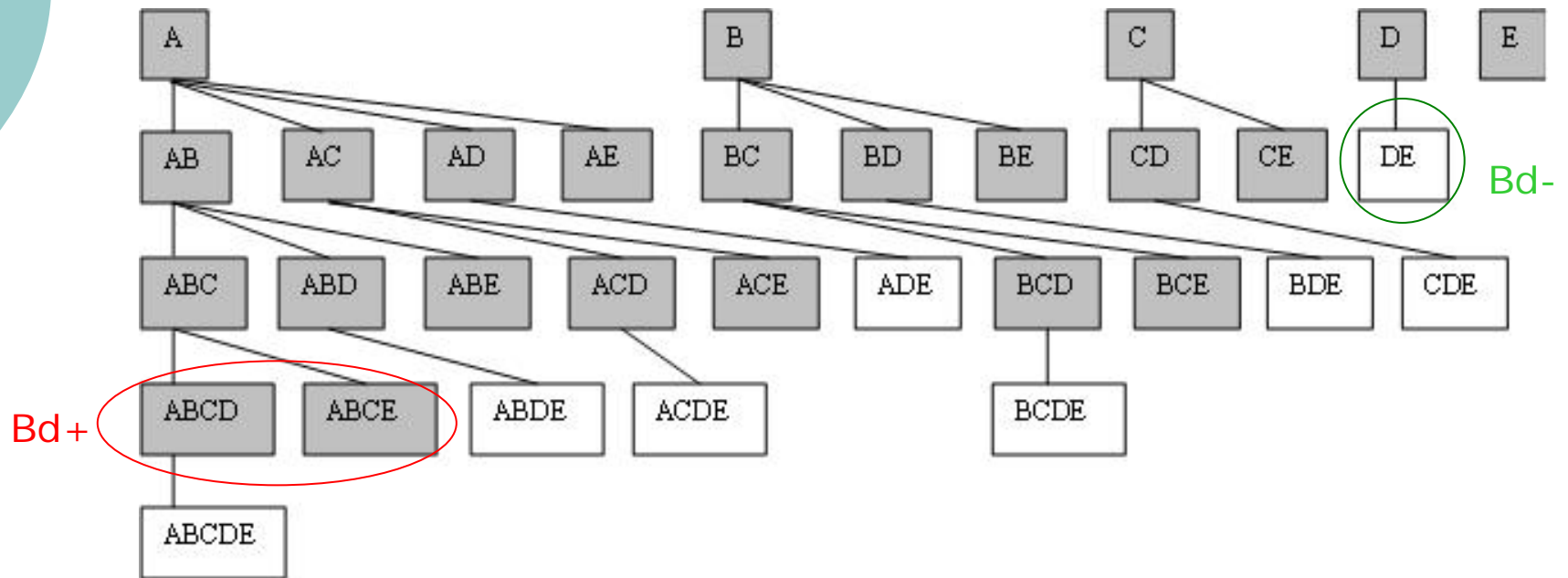
- Devised for maximal frequent itemsets
  - Approach very similar to what we are doing
- but
  - No formal background,
  - No estimate of the error.

# MaxMiner [Sigmod'98]



- Llectic order (Rymon's set enumeration)
- Worst complexity for MaxMiner

# GenMax [ICDM'01]



- Heuristic: “longest pattern containing D or E is of size 4”
- Less precise than our optimistic positive border
- Tradeoff efficiency/ preciseness

# Dualize and Advance [TODS'03]

---

- Randomized algorithm
- Quick overview
  1. From a subset  $I'$  of  $bd^+(I)$ , compute the *optimistic* negative border  $bd^-(I')$
  2. **While** there exists  $X \in bd^-(I')$ ,  $X$  interesting **do**
    - find randomly  $Y \in bd^+(I)$ ,  $Y$  being a maximal interesting specialization of  $X$
    - Go to 1.
- Best theoretical complexity
  - number of dualization (and number of DB scan) very high
- Comparison with our contribution
  - A kind of “symmetric” approach based on a new characterisation presented earlier
  - Different initialization
  - Number of dualization may be tuned

# Conclusion

---

- Characterization of « large candidates » through the simple framework of borders
- Nice setting to apply adaptive strategy
- Perspectives :
  - Maximal frequent itemsets
    - Different initialization
    - More on adaptive behavior
  - Complexity issue