

# Exemplo Algoritmo de consistência de arcos AC-3



Texto base:

David Poole, Alan Mackworth e Randy Goebel -  
*"Computational Intelligence – A logical approach"*

julho/2008

# Algoritmos de consistência

---

- Idéia: podar os domínios tão logo quanto possível, antes de que seus valores sejam selecionados
- Uma variável é **domínio consistente** se nenhum valor do domínio é impossível por qualquer uma das restrições

# Consistência de arcos

---

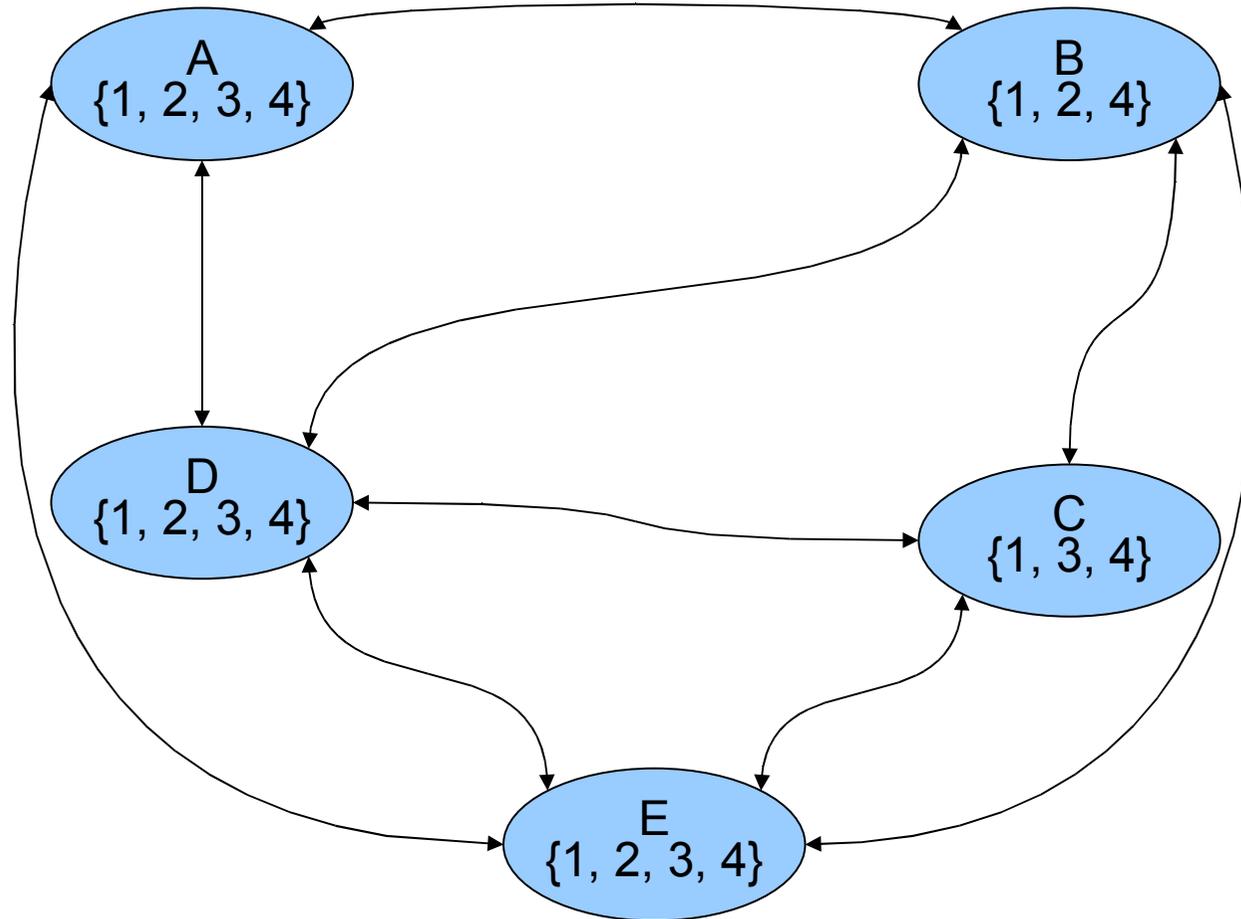
- Uma rede de restrições tem nós correspondentes às variáveis e seus domínios associados
  - Cada relação de restrição  $P(X, Y)$ , corresponde aos arcos  $\langle X, Y \rangle$  e  $\langle Y, X \rangle$
- Um arco  $\langle X, Y \rangle$  é um arco consistente se para cada valor de  $X$  em  $D_x$ , existe algum valor de  $Y$  em  $D_y$ , para o qual a restrição  $P(X, Y)$  é satisfeita
  - Uma rede é arco consistente se todos os seus arcos são arco consistentes

# Exemplo de PSR para escalonamento de atividades

---

- Variáveis: A, B, C, D e E que representam o tempo de início das várias atividades
- Domínios:  $D_A = D_B = D_C = D_D = D_E = \{1, 2, 3, 4, 5\}$
- Restrições:  $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Uma rede de restrições para o problema de escalonamento de atividades



- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

## Algoritmo de consistência de arcos AC-3

---

- Se um arco  $\langle X, Y \rangle$  *não* é arco consistente, todos os valores de  $X$  em  $D_x$  para os quais não existe nenhum valor correspondente em  $D_y$  podem ser apagados de  $D_x$  para fazer  $\langle X, Y \rangle$  arco consistente
- Os arcos são considerados um a um tornando cada um deles consistente
- Um arco  $\langle X, Y \rangle$  necessita ser revisado se o domínio de  $Y$  for reduzido

Entrada:

- um conjunto de variáveis
- um domínio  $D_x$  para cada variável  $X$
- relações  $P_x$  sobre a variável  $X$  que deve ser satisfeita
- relações  $P_{xy}$  sobre as variáveis  $X$  e  $Y$  que deve ser satisfeita

Saída:

- domínios arco consistentes para cada variável

para cada variável  $X$

torne o  $D_x$  consistente de acordo com  $P_x$

TDA = conjunto de arcos a fazer - para cada  $P_{xy}$ ,  $\langle X, Y \rangle$  e  $\langle Y, X \rangle$   
fazem parte do conjunto

repita

selecione qualquer arco de TDA;

exclua este arco de TDA;

Retire todo  $x$  do  $D_x$  que não satisfaça  $P_{xy}$

Se o  $D_x$  foi alterado

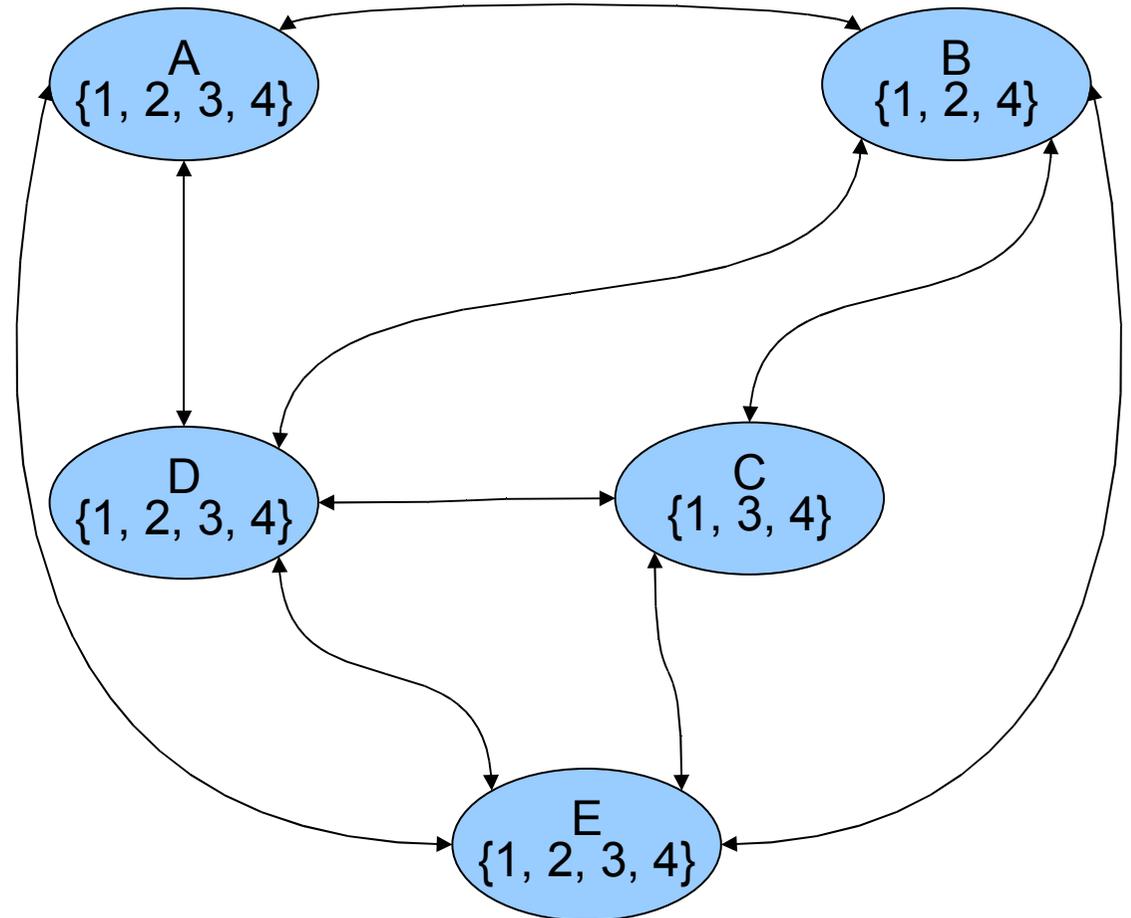
então todos os outros arcos  $\langle Z, X \rangle$  devem ser inseridos em  
TDA

até que TDA seja vazio

# Algoritmo de consistência de arcos AC-3

□ Px:

- $B \neq 3$
- $C \neq 2$



- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

## Entrada:

- um conjunto de variáveis
- um domínio  $D_x$  para cada variável  $X$
- relações  $P_x$  sobre a variável  $X$  que deve ser satisfeita
- relações  $P_{xy}$  sobre as variáveis  $X$  e  $Y$  que deve ser satisfeita

## Saída:

- domínios arco consistentes para cada variável

para cada variável  $X$

torne o  $D_x$  consistente de acordo com  $P_x$

TDA = conjunto de arcos a fazer - para cada  $P_{xy}$ ,  $\langle X, Y \rangle$  e  $\langle Y, X \rangle$   
fazem parte do conjunto

repita

selecione qualquer arco de TDA;

exclua este arco de TDA;

Retire todo  $x$  do  $D_x$  que não satisfaça  $P_{xy}$

Se o  $D_x$  foi alterado

então todos os outros arcos  $\langle Z, X \rangle$  devem ser inseridos em  
TDA

até que TDA seja vazio

# Algoritmo de consistência de arcos AC-3

## □ TDA

1) <A, B>

2) <B, A>

3) <A, D>

4) <D, A>

5) <A, E>

6) <E, A>

7) <B, D>

8) <D, B>

9) <B, C>

10) <C, B>

11) <B, E>

12) <E, B>

13) <D, C>

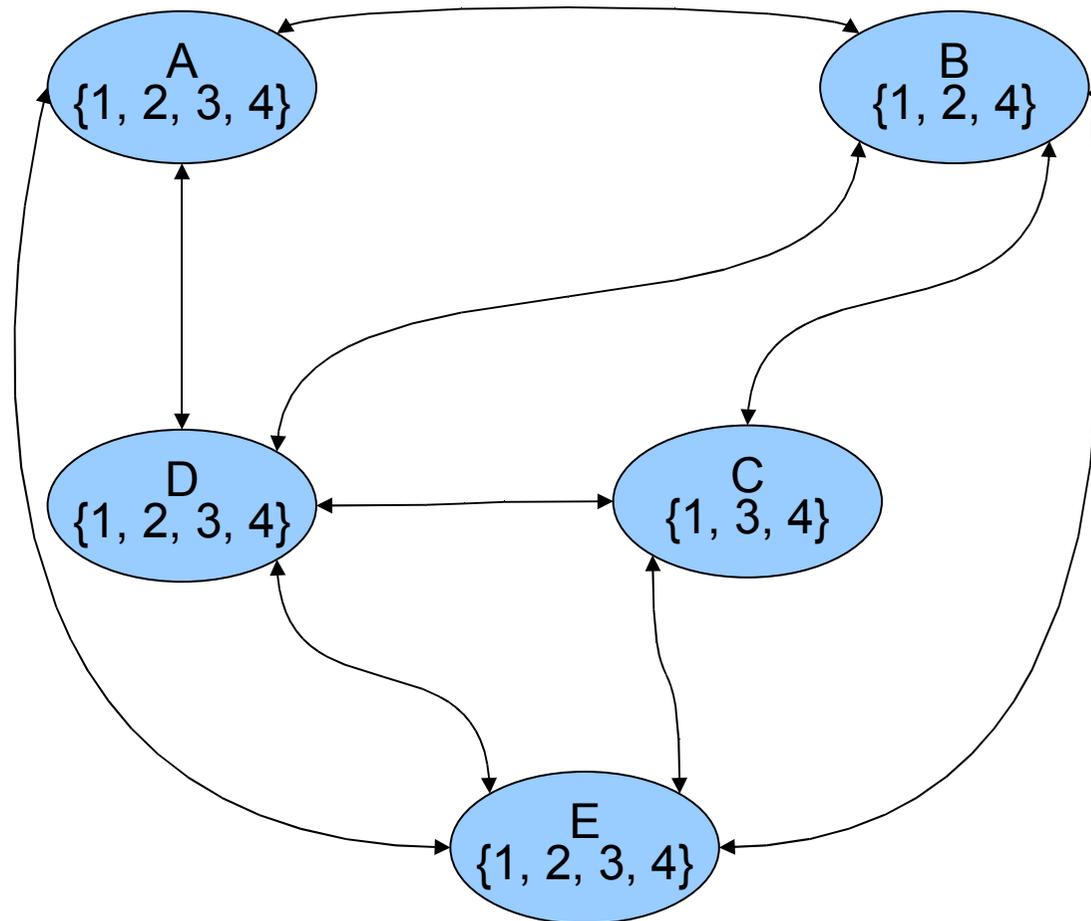
14) <C, D>

15) <D, E>

16) <E, D>

17) <C, E>

18) <E, C>



□  $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

## Entrada:

- um conjunto de variáveis
- um domínio  $D_x$  para cada variável  $X$
- relações  $P_x$  sobre a variável  $X$  que deve ser satisfeita
- relações  $P_{xy}$  sobre as variáveis  $X$  e  $Y$  que deve ser satisfeita

## Saída:

- domínios arco consistentes para cada variável

para cada variável  $X$

torne o  $D_x$  consistente de acordo com  $P_x$

TDA = conjunto de arcos a fazer - para cada  $P_{xy}$ ,  $\langle X, Y \rangle$  e  $\langle Y, X \rangle$  fazem parte do conjunto

repita

selecione qualquer arco de TDA;

exclua este arco de TDA;

Retire todo  $x$  do  $D_x$  que não satisfaça  $P_{xy}$

Se o  $D_x$  foi alterado

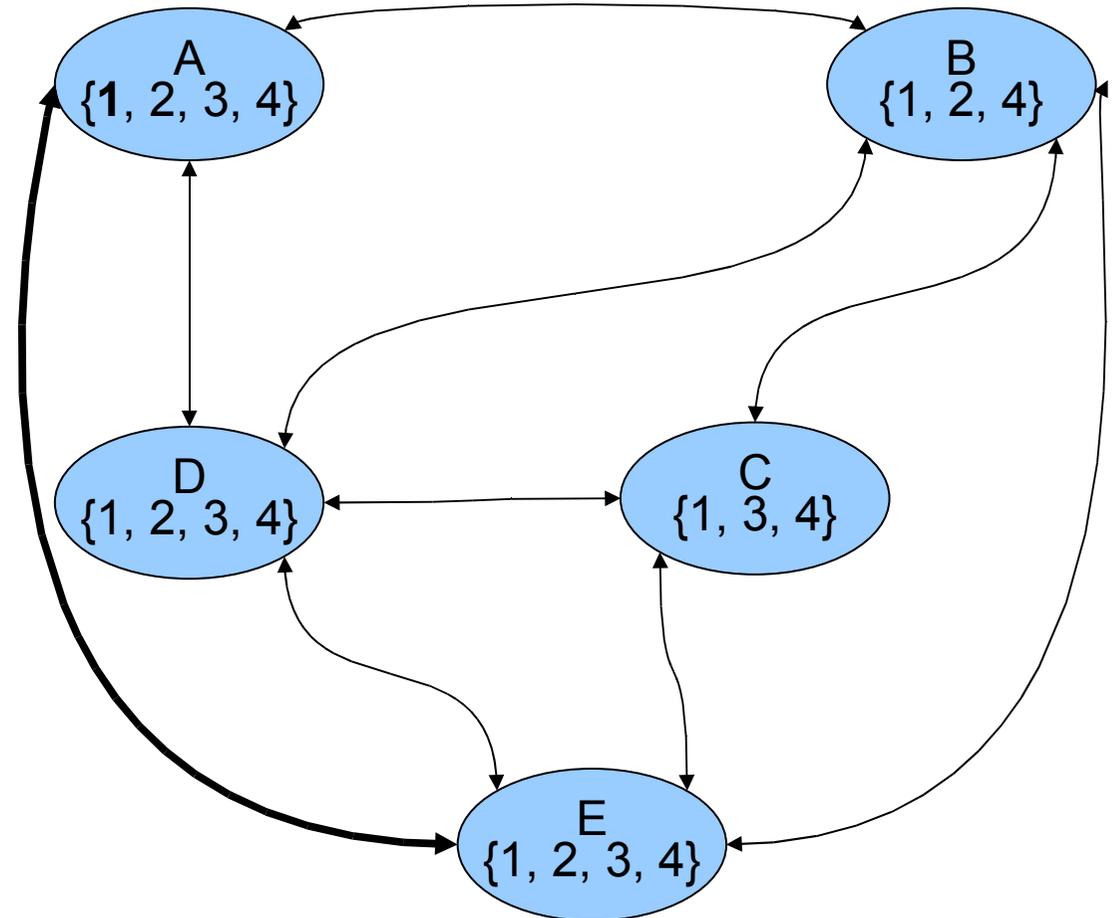
então todos os outros arcos  $\langle Z, X \rangle$  devem ser inseridos em  
TDA

até que TDA seja vazio

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle A, B \rangle$
- ✓  $\langle B, A \rangle$
- ✓  $\langle A, D \rangle$
- ✓  $\langle D, A \rangle$
- ✓  $\langle A, E \rangle$  - tira 1 de A
- ✓  $\langle E, A \rangle$
- ✓  $\langle B, D \rangle$
- ✓  $\langle D, B \rangle$
- ✓  $\langle B, C \rangle$
- ✓  $\langle C, B \rangle$
- ✓  $\langle B, E \rangle$
- ✓  $\langle E, B \rangle$
- ✓  $\langle D, C \rangle$
- ✓  $\langle C, D \rangle$
- ✓  $\langle D, E \rangle$
- ✓  $\langle E, D \rangle$
- ✓  $\langle C, E \rangle$
- ✓  $\langle E, C \rangle$



- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (\mathbf{E < A}) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

✓ <E, A>

✓ <B, D>

✓ <D, B>

✓ <B, C>

✓ <C, B>

✓ <B, E>

✓ <E, B>

✓ <D, C>

✓ <C, D>

✓ <D, E>

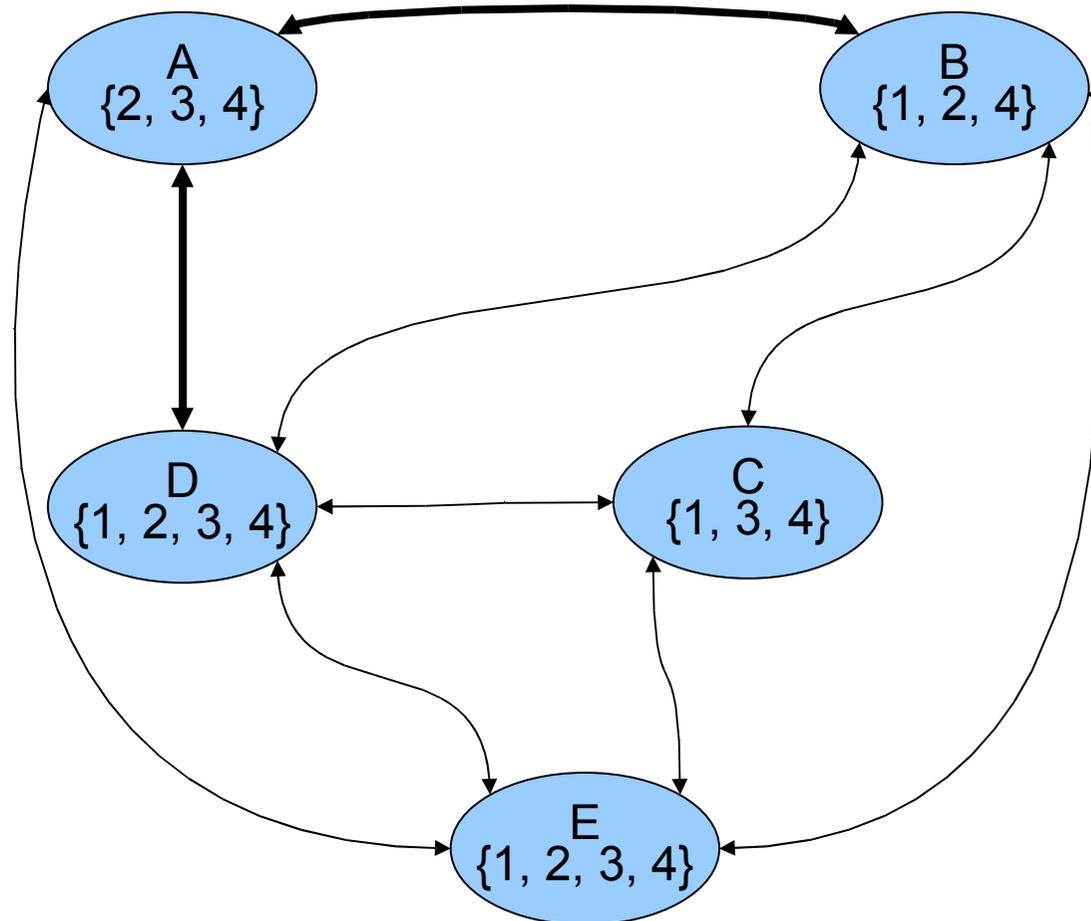
✓ <E, D>

✓ <C, E>

✓ <E, C>

✓ **<B, A>**  $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D)$

✓ **<D, A>**  $\wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$



# Algoritmo de consistência de arcos AC-3

## □ TDA

✓ **<E, A>** - tira 4 de E

✓ <B, D>

✓ <D, B>

✓ <B, C>

✓ <C, B>

✓ <B, E>

✓ <E, B>

✓ <D, C>

✓ <C, D>

✓ <D, E>

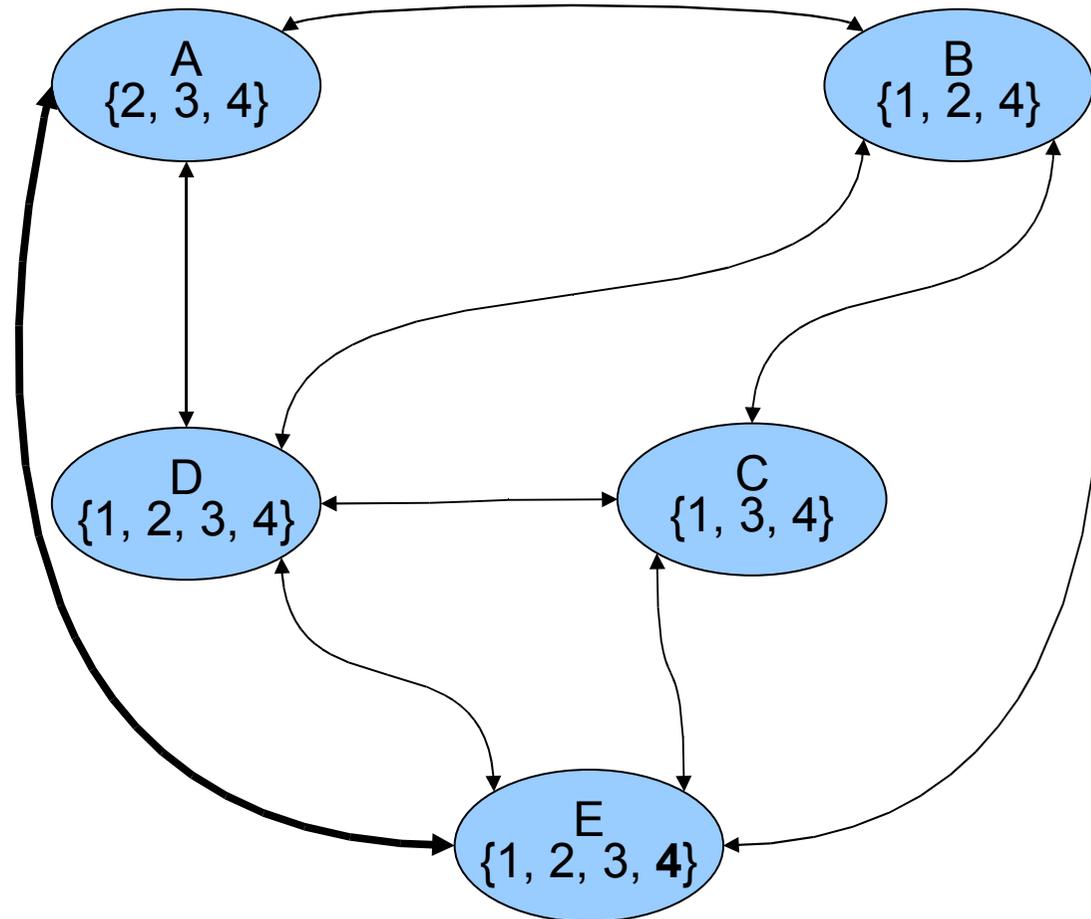
✓ <E, D>

✓ <C, E>

✓ <E, C>

✓ <B, A> □

✓ <D, A>



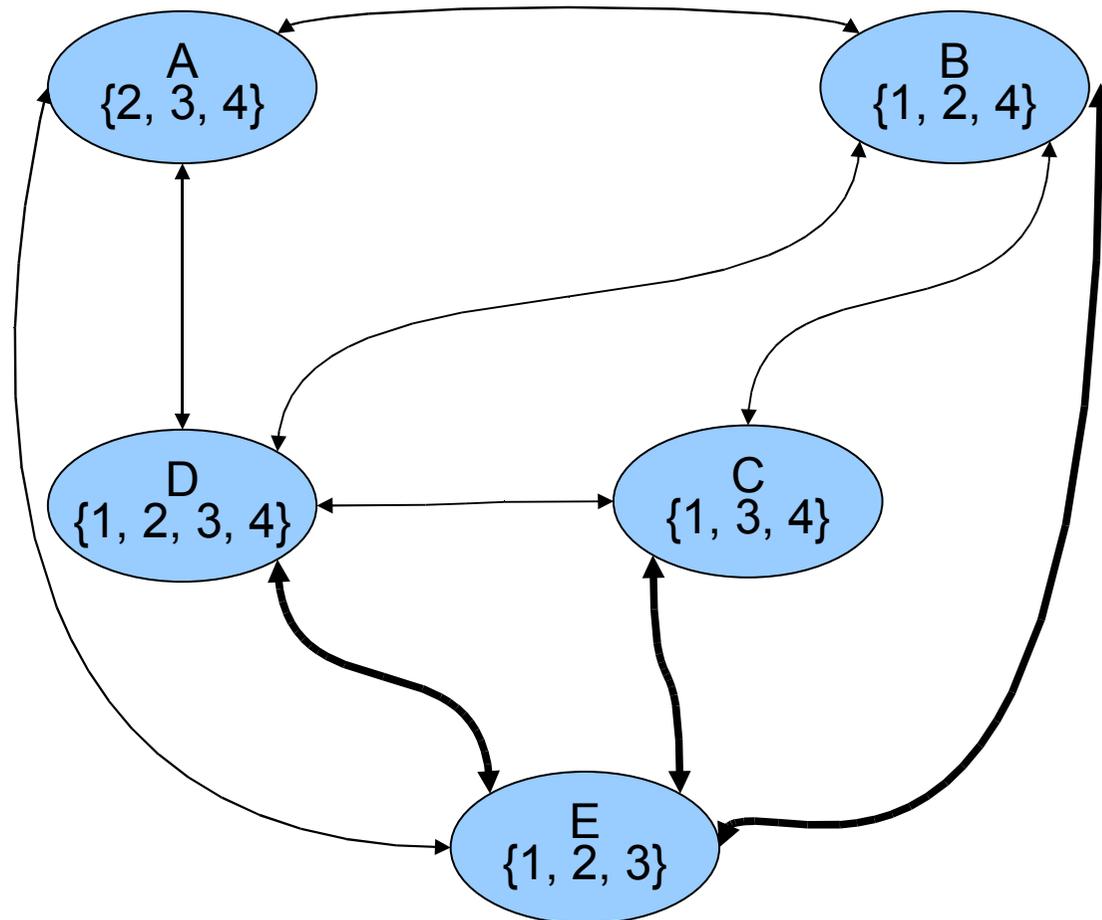
$(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (\mathbf{E < A}) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓ <B, D>
- ✓ <D, B>
- ✓ <B, C>
- ✓ <C, B>
- ✓ **<B, E>**
- ✓ <E, B>

- ✓ <D, C>
- ✓ <C, D>
- ✓ **<D, E>**
- ✓ <E, D>
- ✓ **<C, E>**
- ✓ <E, C>
- ✓ <B, A> □
- ✓ <D, A>



$(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

✓  $\langle B, D \rangle$

✓  $\langle D, B \rangle$

✓  $\langle B, C \rangle$

✓  $\langle C, B \rangle$

✓  $\langle B, E \rangle$  -tira 1 de B

✓  $\langle E, B \rangle$

✓  $\langle D, C \rangle$

✓  $\langle C, D \rangle$

✓  $\langle D, E \rangle$

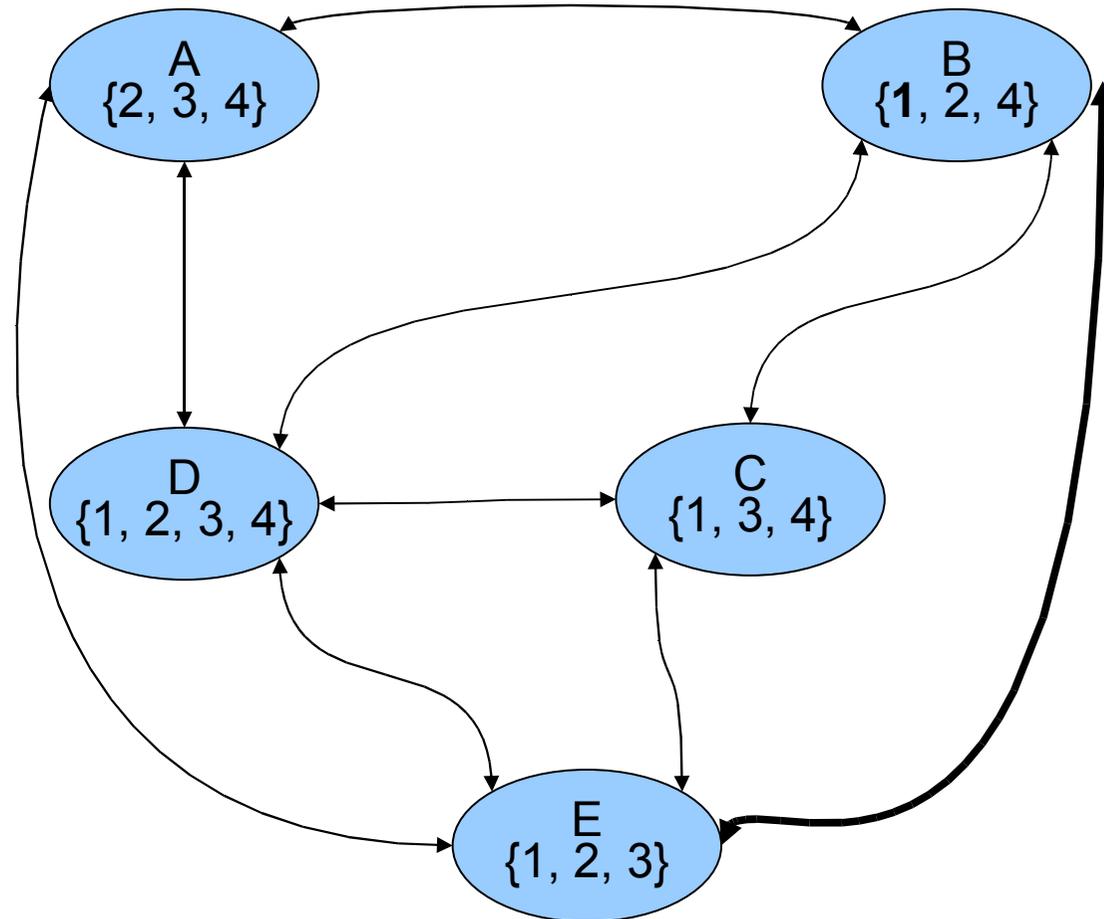
✓  $\langle E, D \rangle$

✓  $\langle C, E \rangle$

✓  $\langle E, C \rangle$

✓  $\langle B, A \rangle$  □

✓  $\langle D, A \rangle$

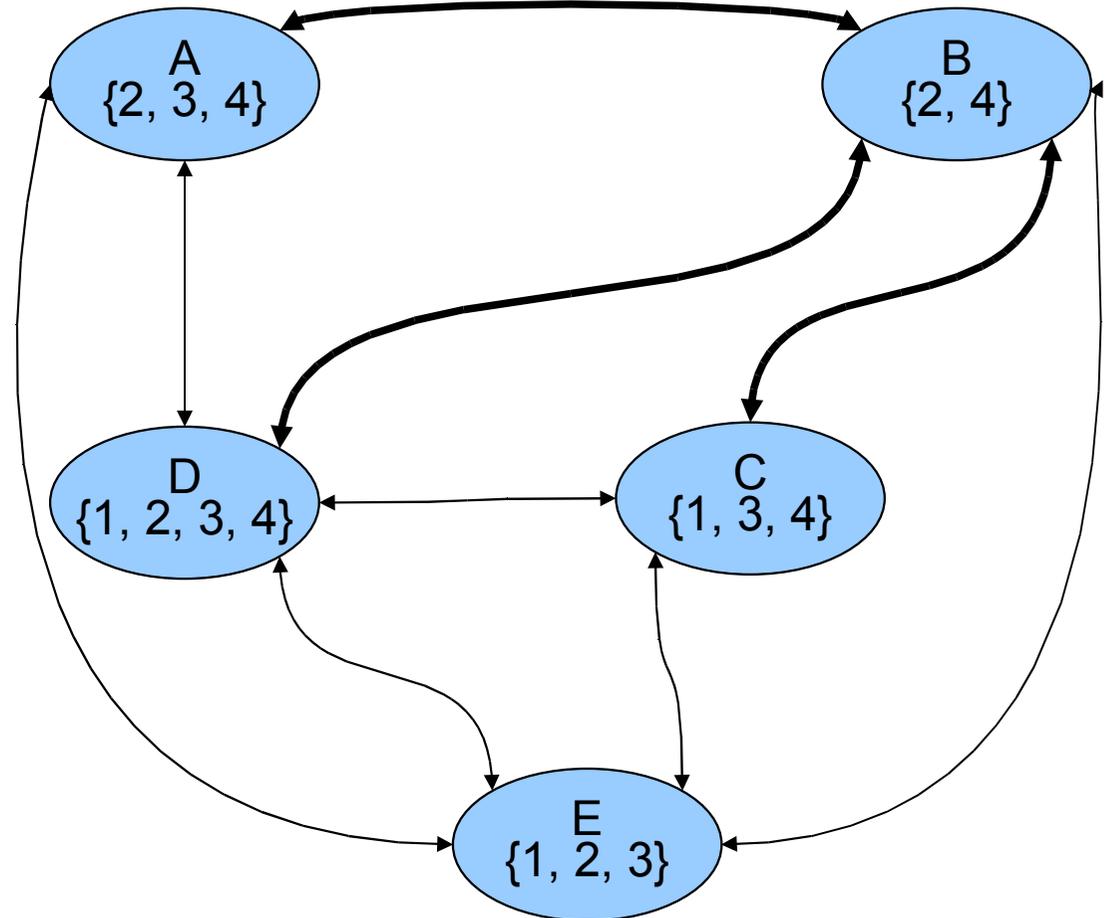


$(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (\mathbf{E < B}) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle E, B \rangle$
- ✓  $\langle D, C \rangle$
- ✓  $\langle C, D \rangle$
- ✓  $\langle D, E \rangle$
- ✓  $\langle E, D \rangle$
- ✓  $\langle C, E \rangle$
- ✓  $\langle E, C \rangle$
- ✓  $\langle B, A \rangle$
- ✓  $\langle D, A \rangle$
- ✓  **$\langle A, B \rangle$**
- ✓  **$\langle D, B \rangle$**
- ✓  **$\langle C, B \rangle$**

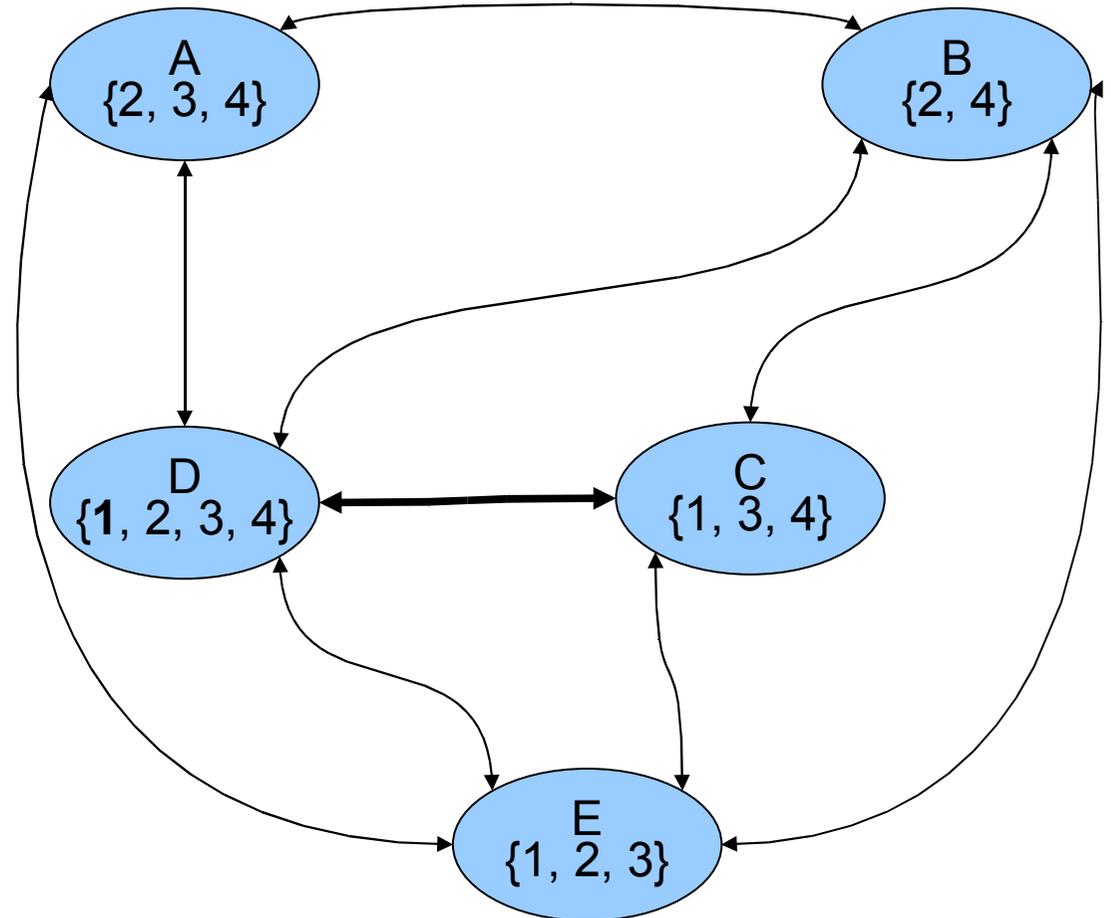


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge \mathbf{(E < B)} \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle E, B \rangle$
- ✓  $\langle D, C \rangle$  - tira 1 de D
- ✓  $\langle C, D \rangle$
- ✓  $\langle D, E \rangle$
- ✓  $\langle E, D \rangle$
- ✓  $\langle C, E \rangle$
- ✓  $\langle E, C \rangle$
- ✓  $\langle B, A \rangle$
- ✓  $\langle D, A \rangle$
- ✓  $\langle A, B \rangle$
- ✓  $\langle D, B \rangle$
- ✓  $\langle C, B \rangle$

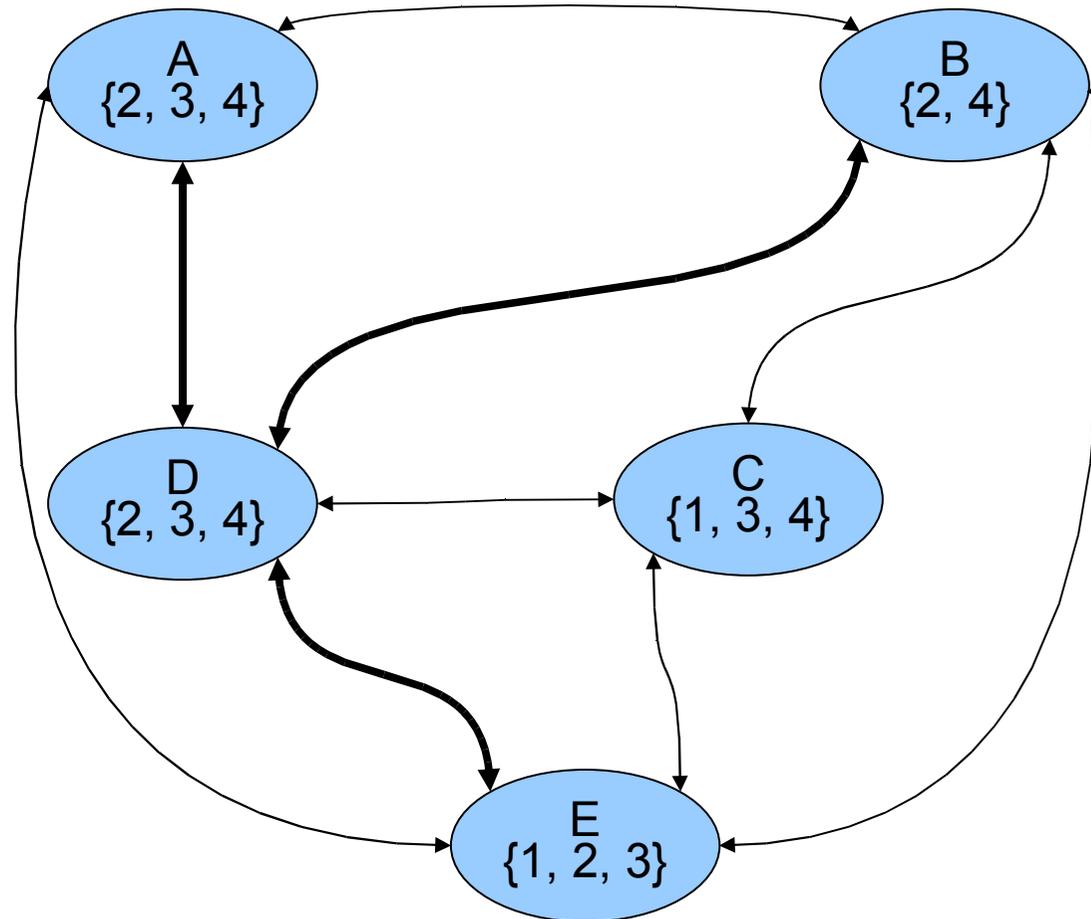


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓ <C, D>
- ✓ <D, E>
- ✓ **<E, D>**
- ✓ <C, E>
- ✓ <E, C>
- ✓ <B, A>
- ✓ <D, A>
- ✓ <A, B>
- ✓ <D, B>
- ✓ <C, B>
- ✓ **<A, D>**
- ✓ **<B, D>**



- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

✓ **<C, D>** - tira 4 de C

✓ <D, E>

✓ <E, D>

✓ <C, E>

✓ <E, C>

✓ <B, A>

✓ <D, A>

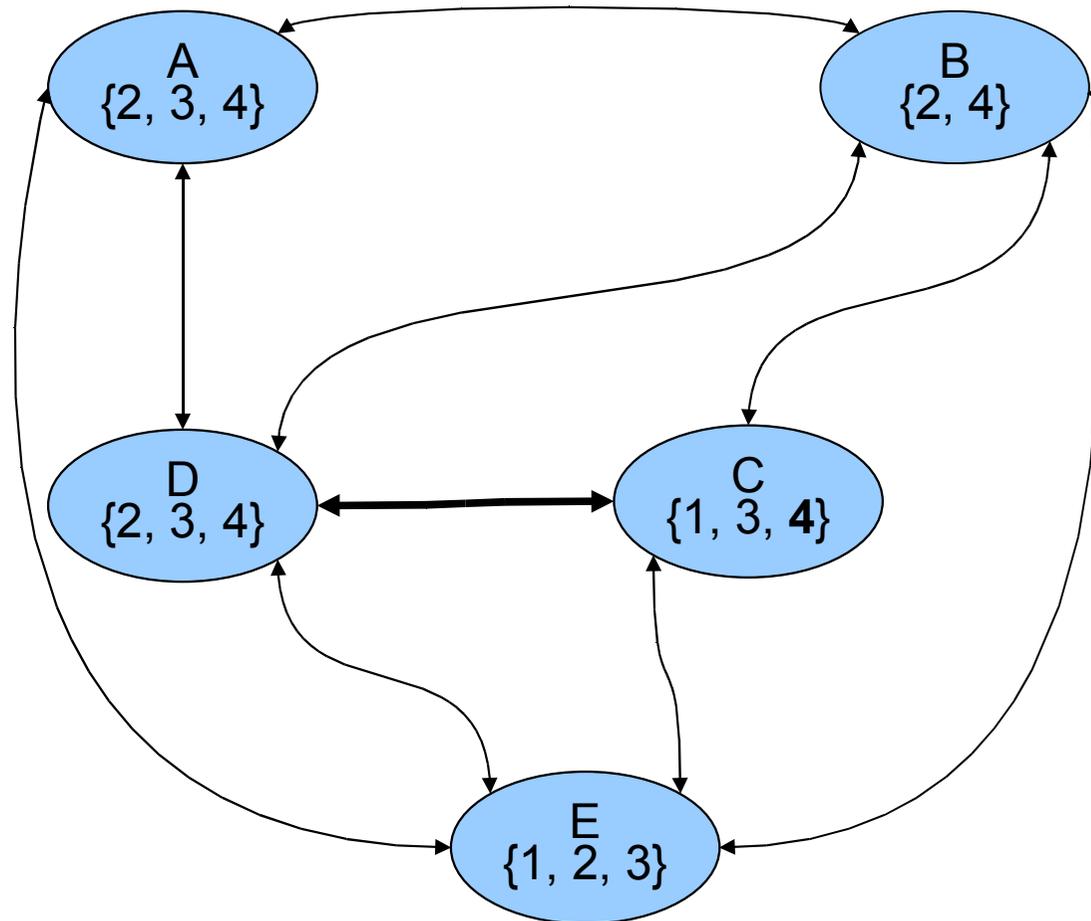
✓ <A, B>

✓ <D, B>

✓ <C, B>

✓ <A, D>

✓ <B, D>

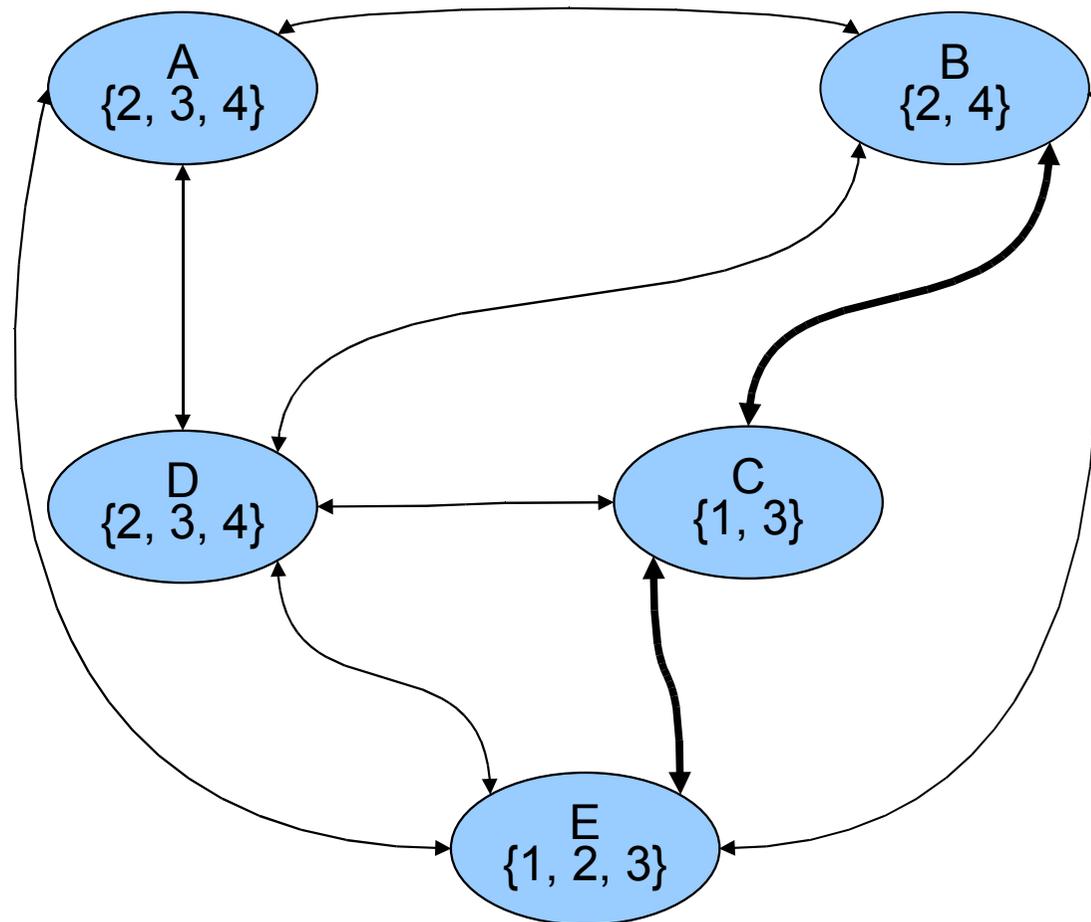


□  $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓ <D, E>
- ✓ <E, D>
- ✓ <C, E>
- ✓ **<E, C>**
- ✓ <B, A>
- ✓ <D, A>
- ✓ <A, B>
- ✓ <D, B>
- ✓ <C, B>
- ✓ <A, D>
- ✓ <B, D>
- ✓ **<B, C>**

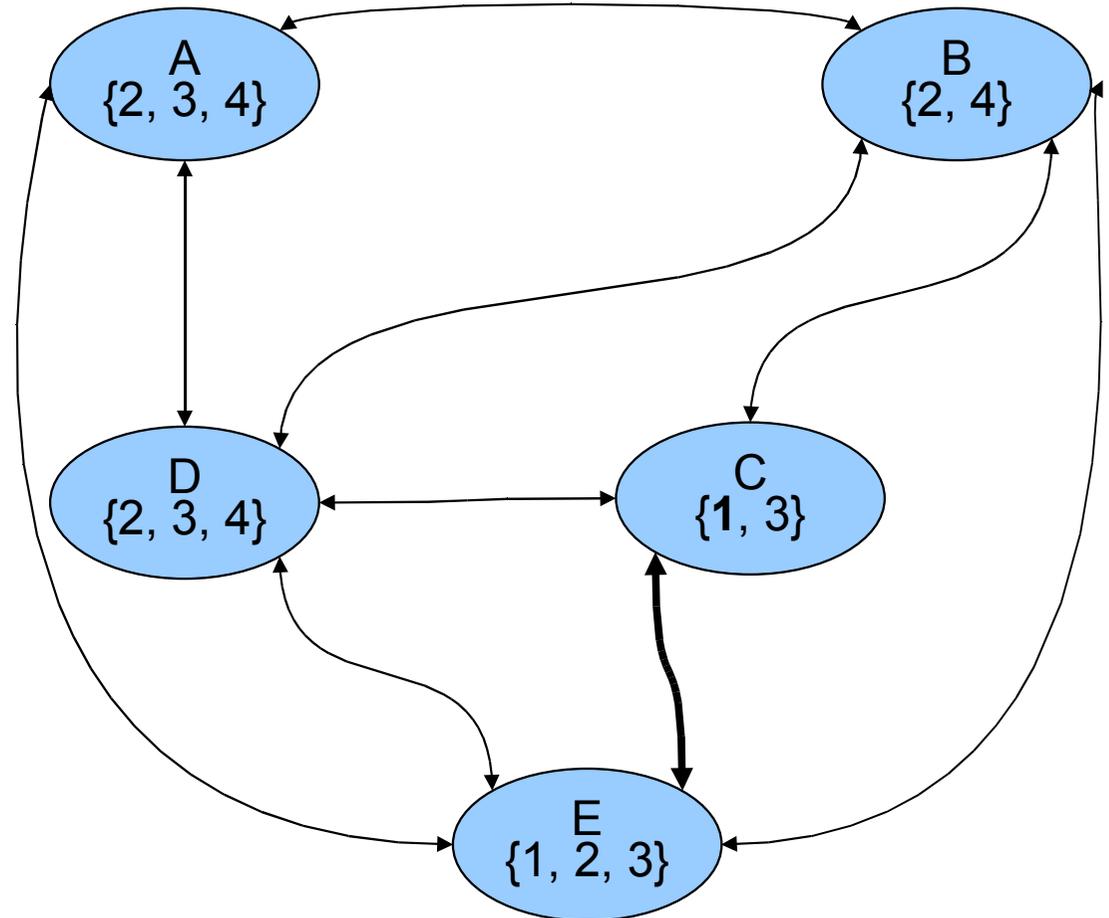


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle D, E \rangle$
- ✓  $\langle E, D \rangle$
- ✓  $\langle C, E \rangle$  - tira 1 de C
- ✓  $\langle E, C \rangle \langle B, A \rangle$
- ✓  $\langle D, A \rangle$
- ✓  $\langle A, B \rangle$
- ✓  $\langle D, B \rangle$
- ✓  $\langle C, B \rangle$
- ✓  $\langle A, D \rangle$
- ✓  $\langle B, D \rangle$
- ✓  $\langle B, C \rangle$

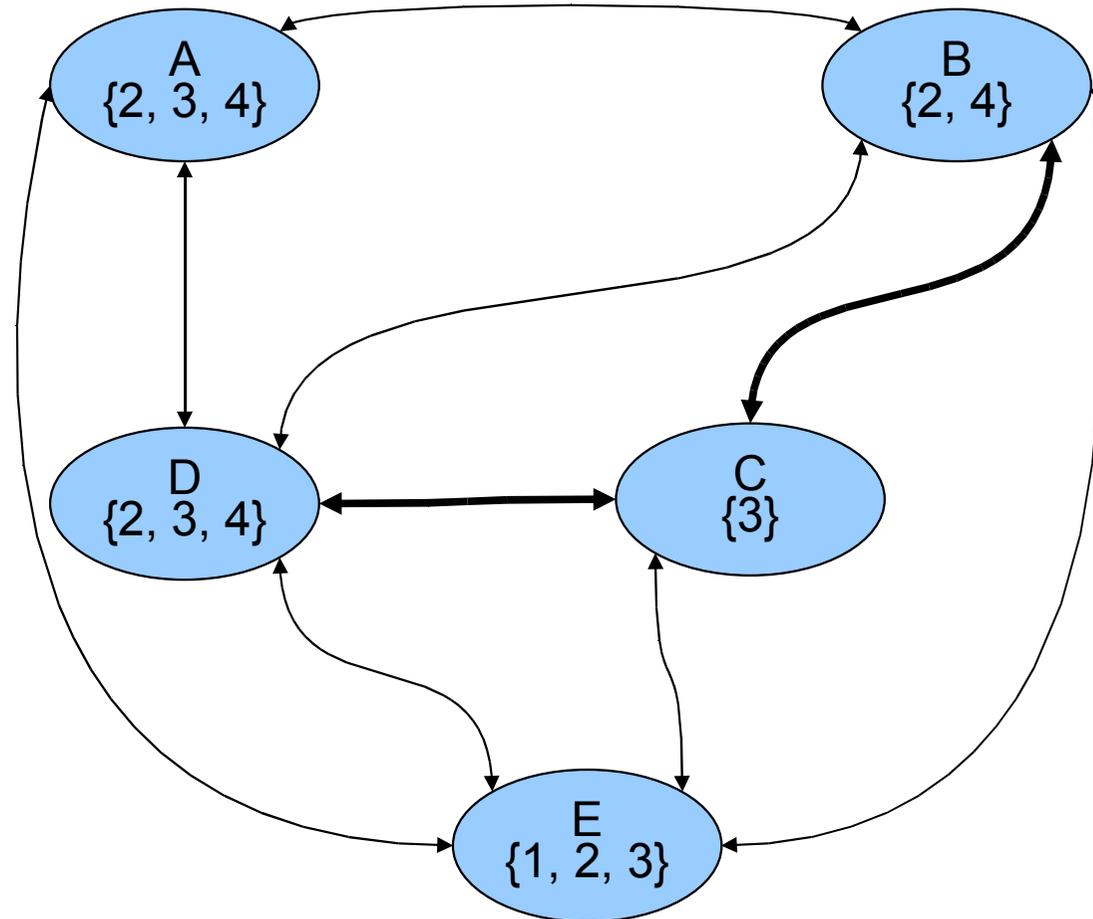


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓ <E, C>
- ✓ <B, A>
- ✓ <D, A>
- ✓ <A, B>
- ✓ <D, B>
- ✓ <C, B>
- ✓ <A, D>
- ✓ <B, D>
- ✓ **<B, C>**
- ✓ **<D, C>**

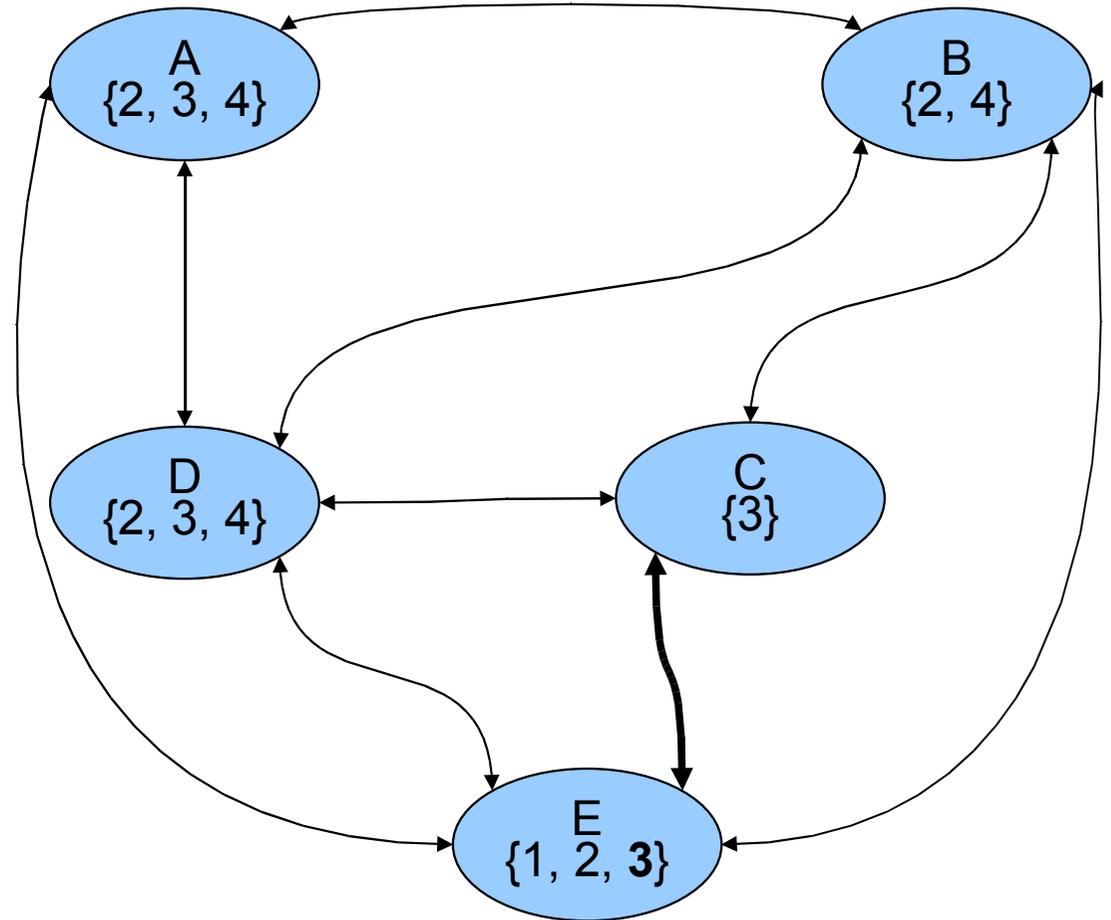


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓ **<E, C>** - tira 3 de E
- ✓ <B, A>
- ✓ <D, A>
- ✓ <A, B>
- ✓ <D, B>
- ✓ <C, B>
- ✓ <A, D>
- ✓ <B, D>
- ✓ <B, C>
- ✓ <D, C>

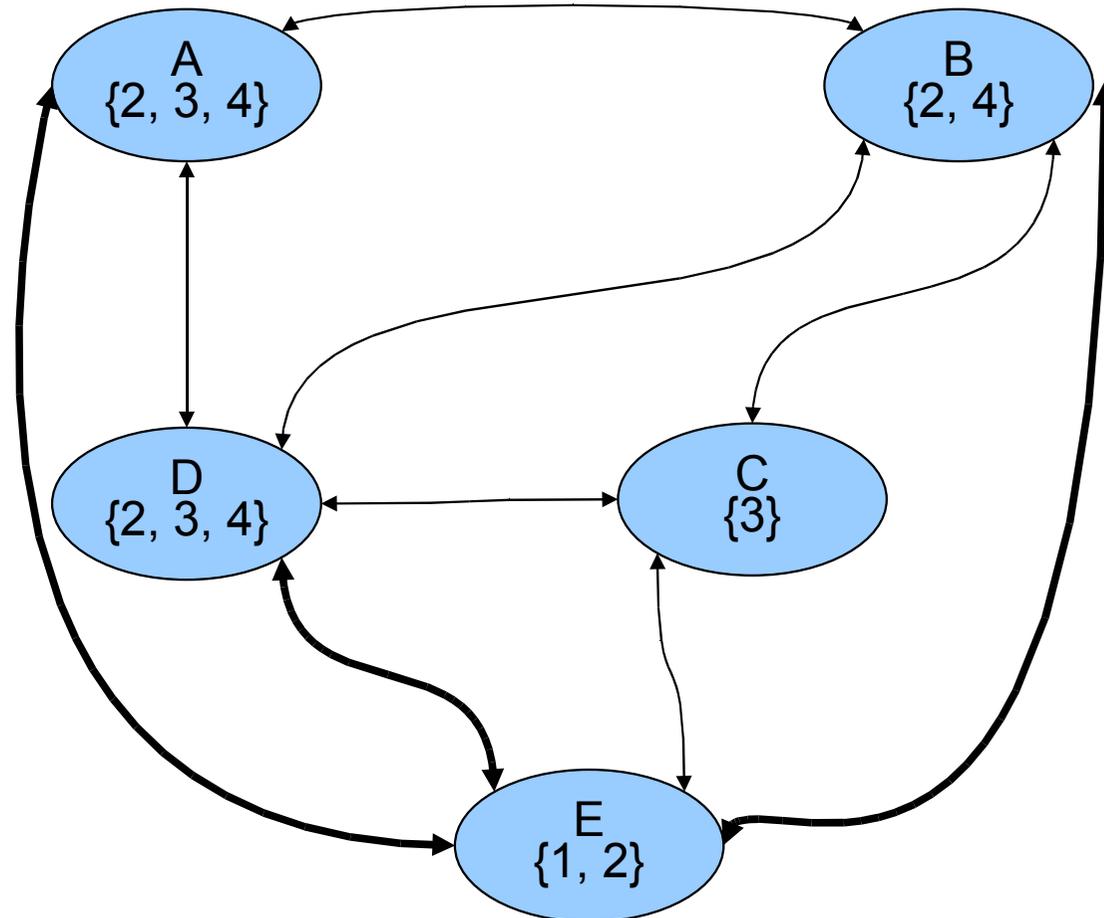


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge \mathbf{(E < C)} \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle B, A \rangle$
- ✓  $\langle D, A \rangle$
- ✓  $\langle A, B \rangle$
- ✓  $\langle D, B \rangle$
- ✓  $\langle C, B \rangle$
- ✓  $\langle A, D \rangle$
- ✓  $\langle B, D \rangle$
- ✓  $\langle B, C \rangle$
- ✓  $\langle D, C \rangle$
- ✓  **$\langle A, E \rangle$**
- ✓  **$\langle B, E \rangle$**
- ✓  **$\langle D, E \rangle$**

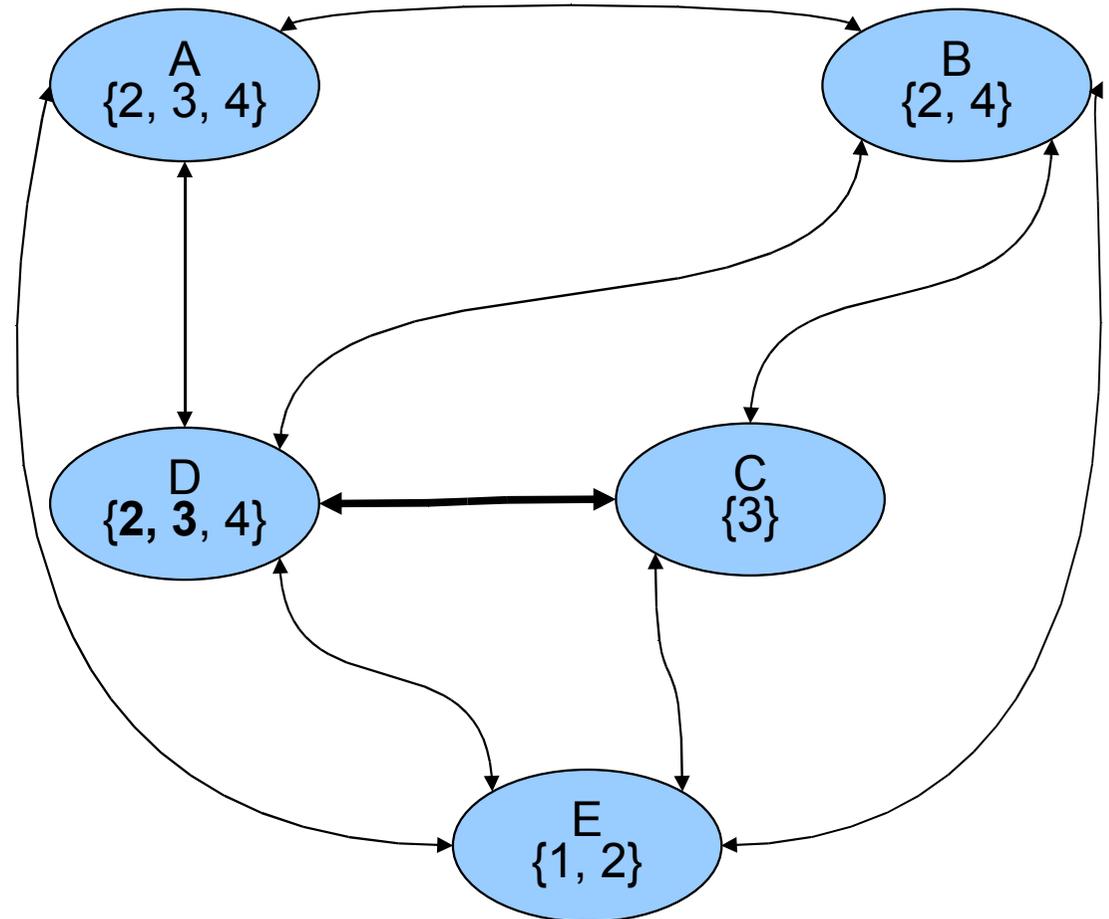


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle B, A \rangle$
- ✓  $\langle D, A \rangle$
- ✓  $\langle A, B \rangle$
- ✓  $\langle D, B \rangle$
- ✓  $\langle C, B \rangle$
- ✓  $\langle A, D \rangle$
- ✓  $\langle B, D \rangle$
- ✓  $\langle B, C \rangle$
- ✓  $\langle D, C \rangle$  - tira 2 e 3 de D
- ✓  $\langle A, E \rangle$
- ✓  $\langle B, E \rangle$
- ✓  $\langle D, E \rangle$

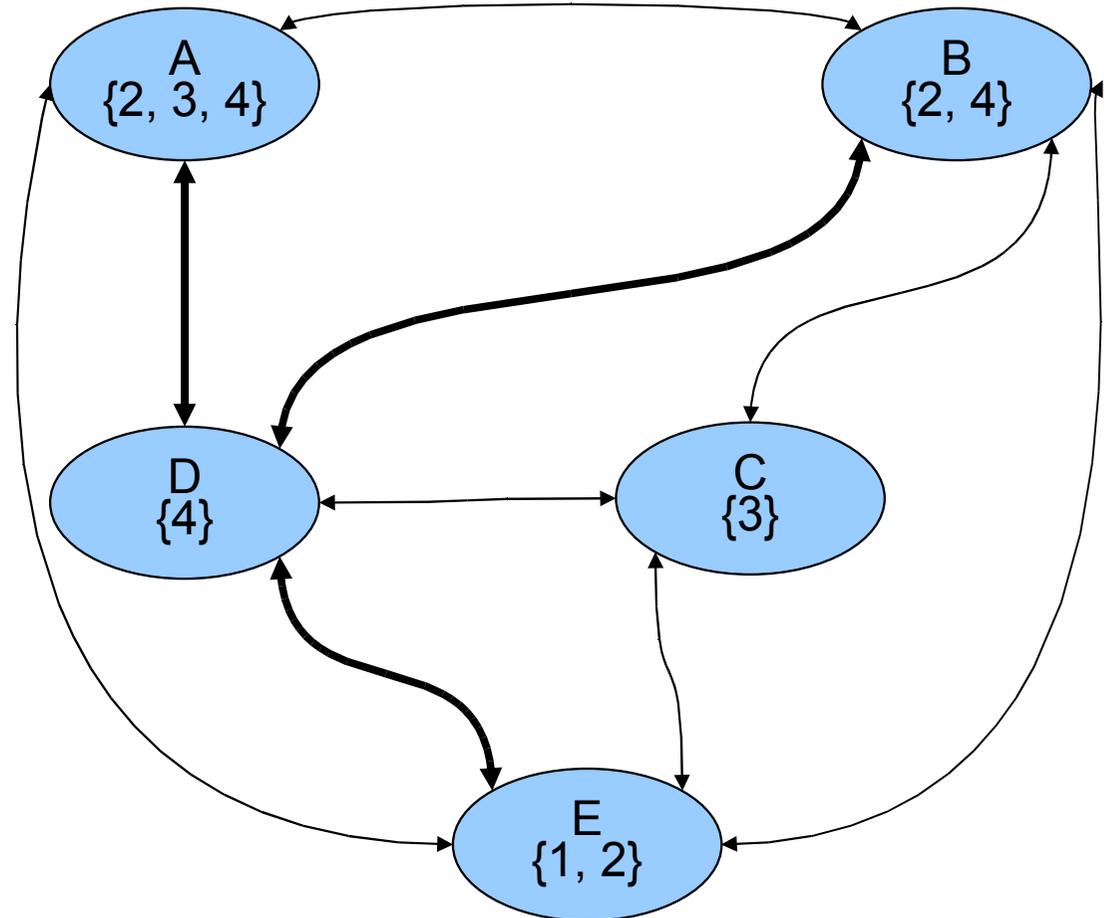


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle A, E \rangle$
- ✓  $\langle B, E \rangle$
- ✓  $\langle D, E \rangle$
- ✓  $\langle \mathbf{A}, \mathbf{D} \rangle$
- ✓  $\langle \mathbf{B}, \mathbf{D} \rangle$
- ✓  $\langle \mathbf{E}, \mathbf{D} \rangle$

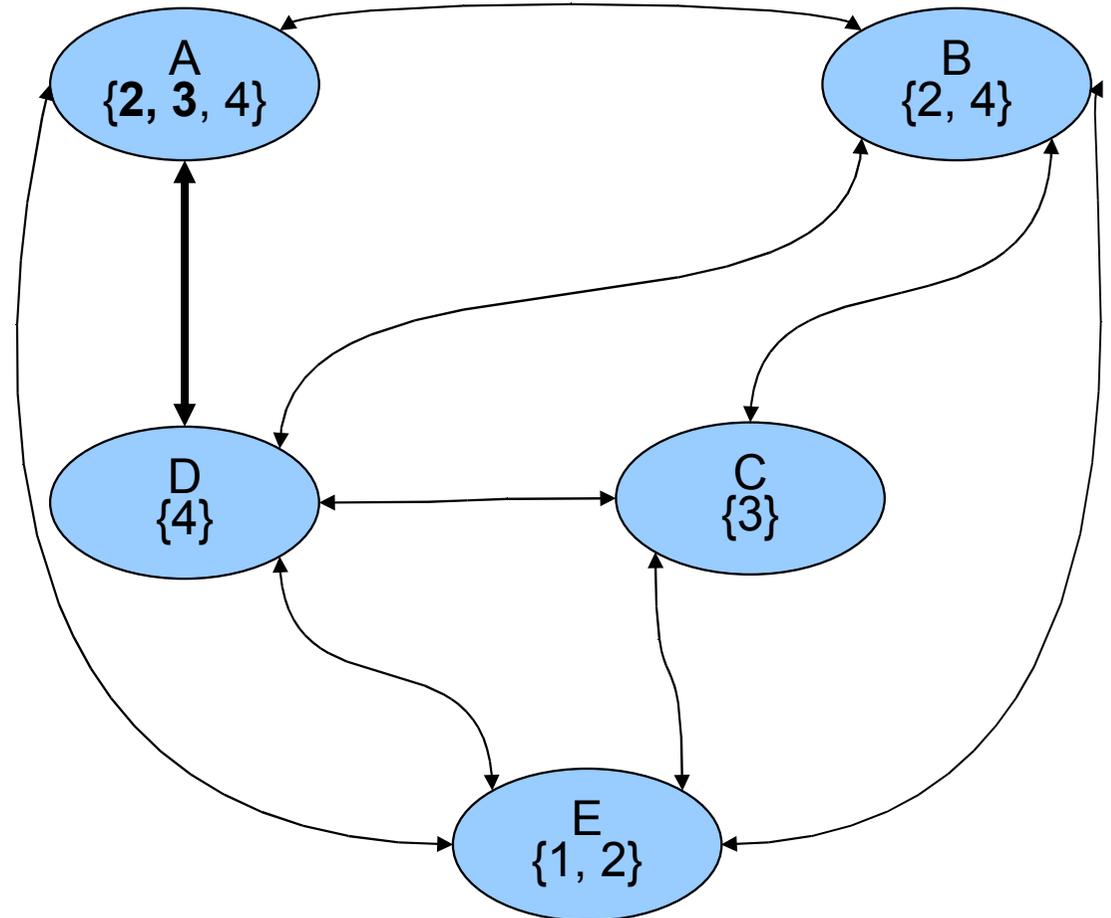


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle A, E \rangle$
- ✓  $\langle B, E \rangle$
- ✓  $\langle D, E \rangle$
- ✓  $\langle A, D \rangle$  - tira 2 e 3 de A
- ✓  $\langle B, D \rangle$
- ✓  $\langle E, D \rangle$

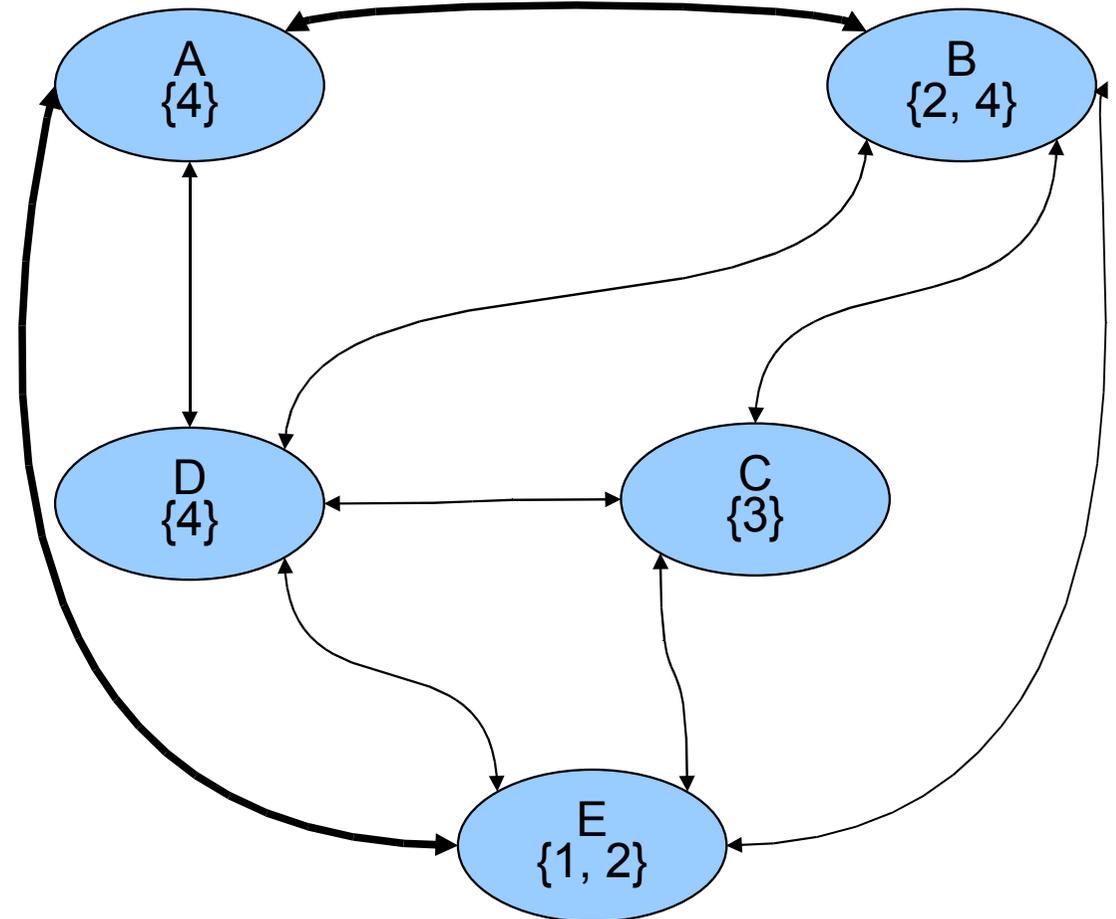


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle B, D \rangle$
- ✓  $\langle E, D \rangle$
- ✓  $\langle \mathbf{B}, \mathbf{A} \rangle$
- ✓  $\langle \mathbf{E}, \mathbf{A} \rangle$

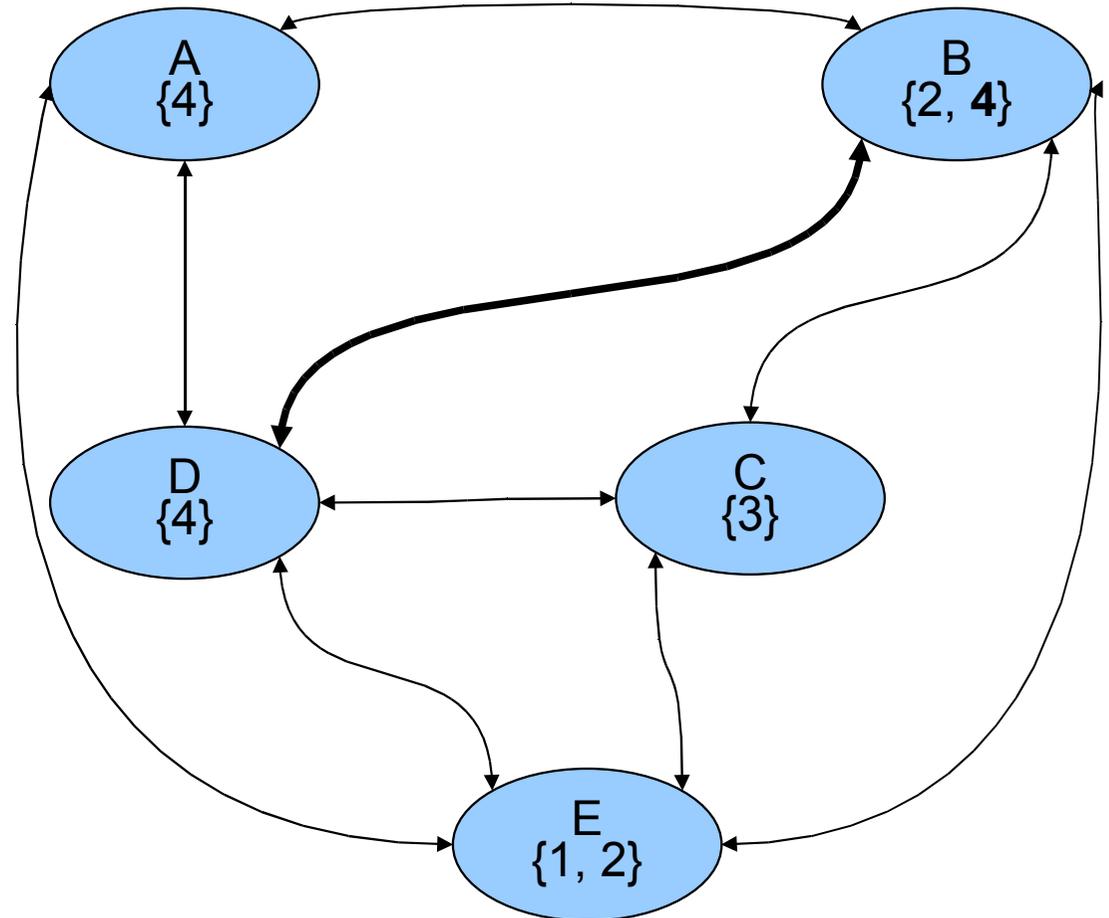


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓ **<B, D>** - tira 4 de B
- ✓ <E, D>
- ✓ <B, A>
- ✓ <E, A>

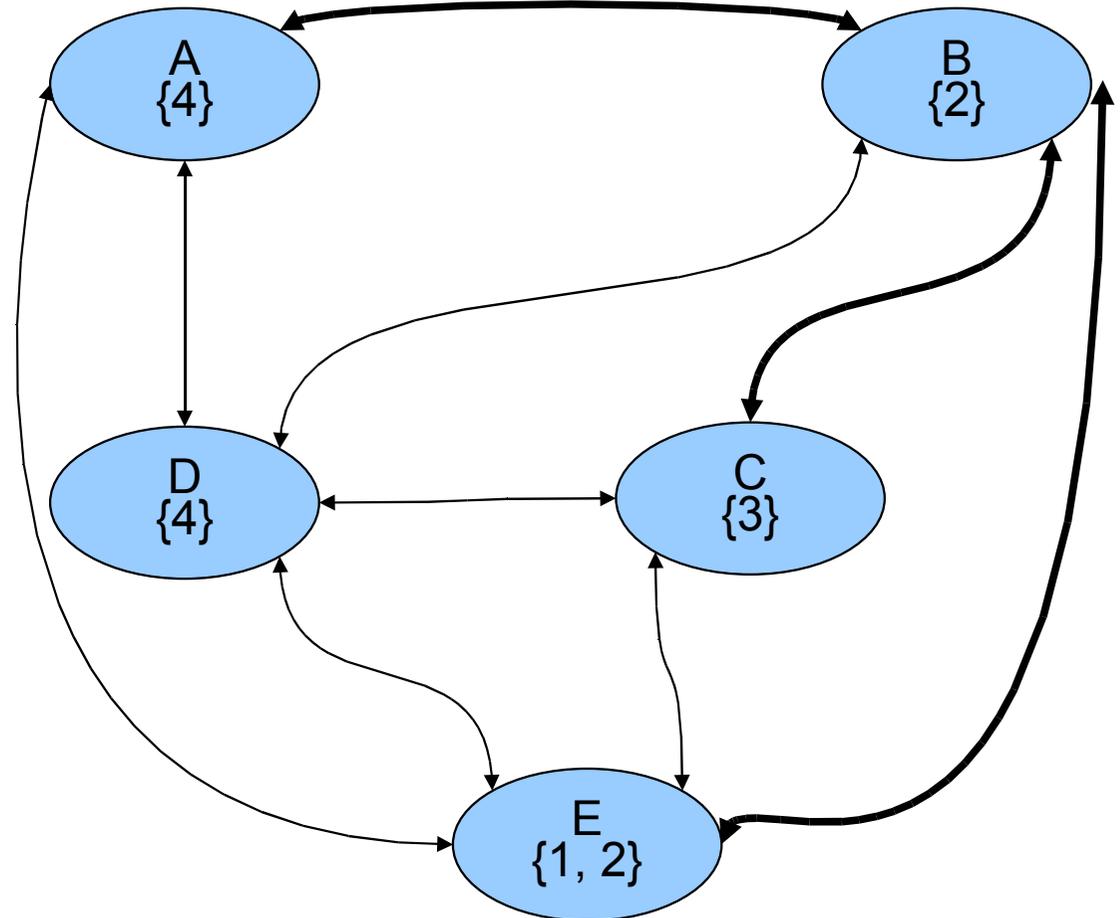


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge \mathbf{(B \neq D)} \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle E, D \rangle$
- ✓  $\langle B, A \rangle$
- ✓  $\langle E, A \rangle$
- ✓  **$\langle A, B \rangle$**
- ✓  **$\langle C, B \rangle$**
- ✓  **$\langle E, B \rangle$**

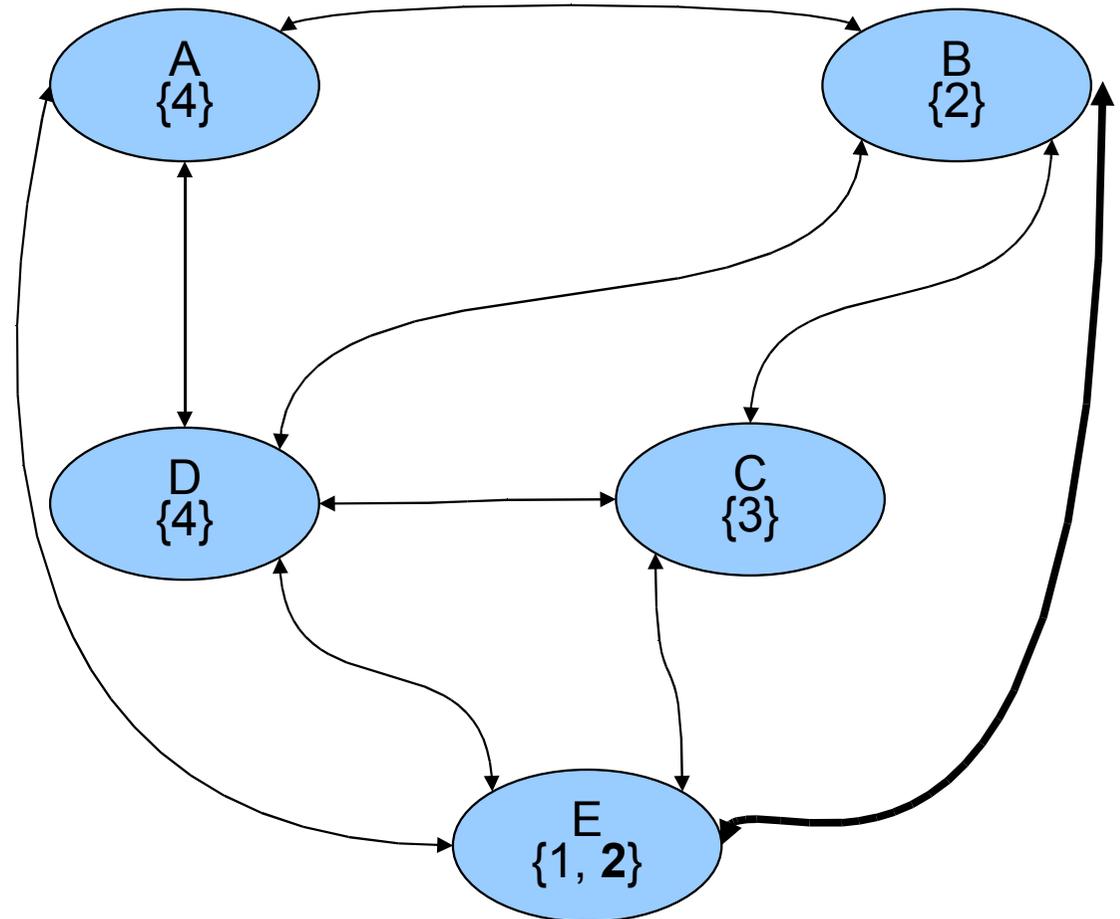


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle E, D \rangle$
- ✓  $\langle B, A \rangle$
- ✓  $\langle E, A \rangle$
- ✓  $\langle A, B \rangle$
- ✓  $\langle C, B \rangle$
- ✓  $\langle E, B \rangle$  - tira 2 de E

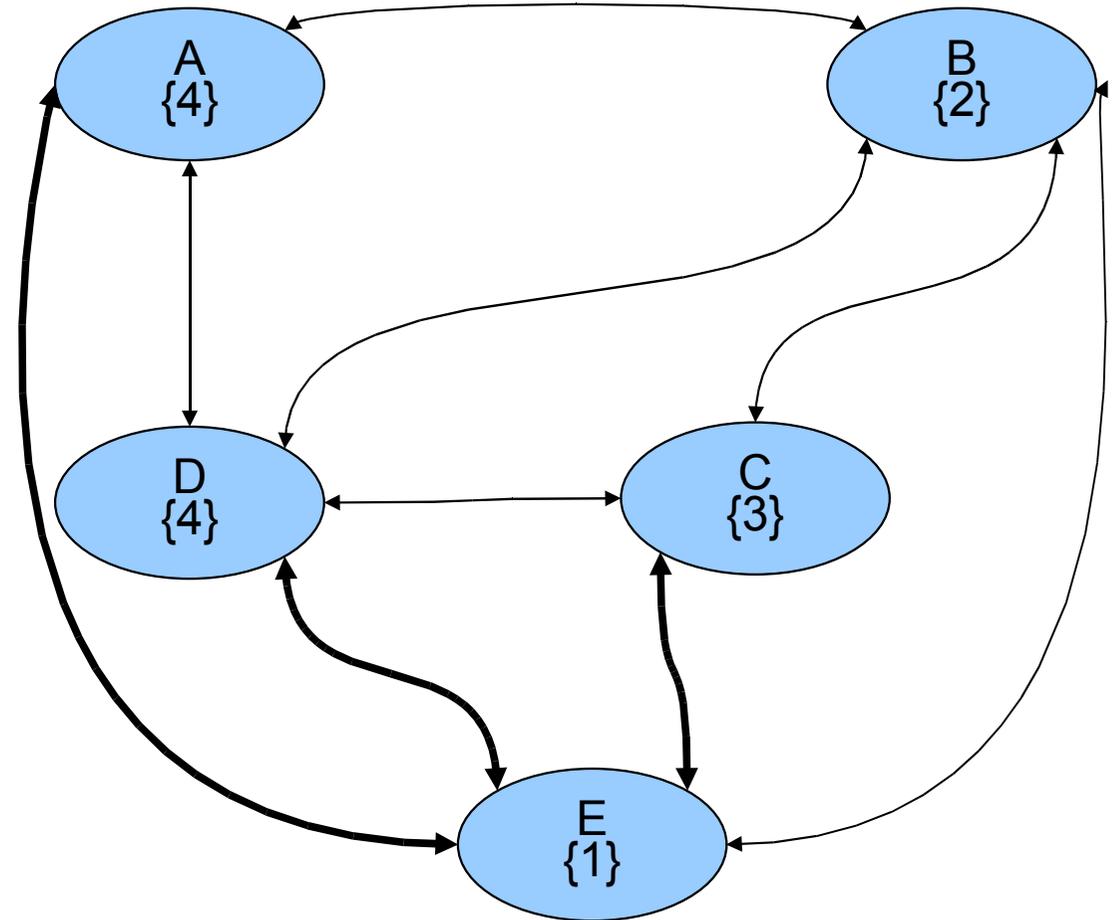


- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (\mathbf{E < B}) \wedge (E < C) \wedge (E < D)$

# Algoritmo de consistência de arcos AC-3

## □ TDA

- ✓  $\langle A, E \rangle$
- ✓  $\langle D, E \rangle$
- ✓  $\langle C, E \rangle$



- $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

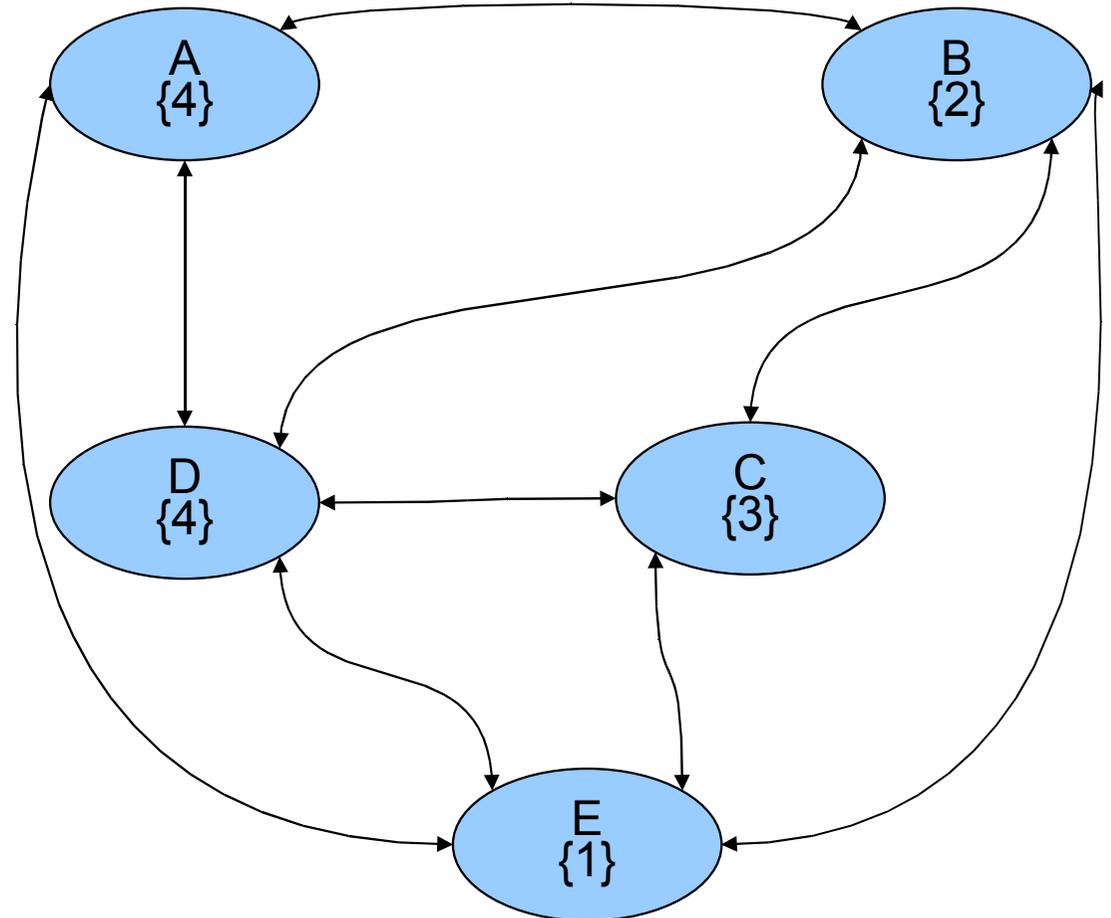
# Algoritmo de consistência de arcos AC-3

## □ TDA

✓  $\langle A, E \rangle$

✓  $\langle D, E \rangle$

✓  $\langle C, E \rangle$



□  $(B \neq 3) \wedge (C \neq 2) \wedge (A \neq B) \wedge (B \neq C) \wedge (B \neq D) \wedge (C < D) \wedge (A = D) \wedge (E < A) \wedge (E < B) \wedge (E < C) \wedge (E < D)$

# Resultados do algoritmo AC-3

---

- Um domínio é vazio
  - Não existe solução para o problema
- Cada domínio tem somente um valor
  - Existe somente uma solução para o problema
- Alguns domínios tem mais que um valor
  - Dividir o domínio e aplicar recursivamente o AC-3
  - É melhor dividir o menor domínio
  - Inicialmente todos os arcos são arco-consistentes exceto aqueles que apontam para o domínio que foi dividido
- Se cada variável tem domínio de tamanho  $d$  e  $e$  relações a serem testadas, AC-3 tem complexidade  $O(ed^3)$ 
  - Para grafos de restrições em forma de árvore AC-3 é linear no número de variáveis