

Presburger Arithmetic

- Which arithmetical problems can be solved using automata?
- Presburger arithmetic: a logical language to define properties of (tuples of) natural numbers

Syntax of PA

- Symbols: variables $x, y, z \dots$
 constants $0, 1$
 arithmetic symbols $+, =, <$
 logical symbols or, not, Exists
 parenthesis
- Terms: a variable is a term
 0 and 1 are terms
 if t and u are terms, then $t + u$ is a term
- Atomic formulas: $t = < u$, where t and u are terms

Syntax of PA

- Formulas:
 - every atomic formula is a formula;
 - if φ_1, φ_2 are formulas, then so are $\neg\varphi_1$, $\varphi_1 \vee \varphi_2$, and $\exists x\varphi_1$.
- Free and bound variables:
 - a variable is bound if it is in the scope of an existential quantifier, otherwise it is free.
- A formula without free variables is called a sentence

Abbreviations

Conjunction, implication, bi-implication, universal quantification

$$\begin{array}{ll}
 n & = \underbrace{1 + 1 + \dots + 1}_{n \text{ times}} \\
 nx & = \underbrace{x + x + \dots + x}_{n \text{ times}} \\
 t \geq t' & = t' \leq t \\
 t = t' & = t \leq t' \wedge t \geq t' \\
 t < t' & = t \leq t' \wedge \neg(t = t') \\
 t > t' & = t' < t
 \end{array}$$

Semantics (intuition)

- The semantics of a sentence is "true" or "false"
- The semantics of a formula with free variables ($x_1, \dots, x_k$) is the set containing all tuples $(n_1, \dots, n_k)$ of natural numbers that "satisfy the formula"

Semantics (more formally)

- An interpretation of a formula F is any function that assigns a natural number to every variable appearing in f (and perhaps also to others).

Given an interpretation I , a variable x , and a number n , we denote by $I[n/x]$ the interpretation that assigns to x the number n , and to all other variables the same value as I .

Semantics (more formally)

- We define when an interpretation satisfies a formula F .

$$\mathcal{I} \models t \leq u \quad \text{iff} \quad \mathcal{I}(t) \leq \mathcal{I}(u)$$

$$\mathcal{I} \models \neg \varphi_1 \quad \text{iff} \quad \mathcal{I} \not\models \varphi_1$$

$$\mathcal{I} \models \varphi_1 \vee \varphi_2 \quad \text{iff} \quad \mathcal{I} \models \varphi_1 \text{ or } \mathcal{I} \models \varphi_2$$

$$\mathcal{I} \models \exists x \varphi \quad \text{iff} \quad \text{there exists } n \geq 0 \text{ such that } \mathcal{I}[n/x] \models \varphi$$

- Lemma: Let F be a formula, and let $\mathcal{I}_1, \mathcal{I}_2$ be two interpretations of F . If $\mathcal{I}_1$ and $\mathcal{I}_2$ assign the same values to all FREE variables of F , then either they both satisfy F or none of them satisfies F .
- Consequence: if F is a sentence, either all interpretations satisfy F , or none of them satisfies F .

Semantics (more formally)

- We say a sentence is true if it is satisfied by all interpretations.
- We say a sentence is false if it is not satisfied by any interpretation.
- A model or solution of a formula F is the projection of any interpretation that satisfies F onto the free variables of F .
- The set of models or solutions of F is also called the solution space of F , and denoted by $\text{Sol}(F)$.

Language of a formula

we encode natural numbers as strings over $\{0, 1\}$ using the least-significant-bit-first encoding *lsbf*. If we have free variables $x_1, \dots, x_k$, the elements of the solution space are encoded as a word over $\{0, 1\}^k$. For instance, the word

$$\begin{array}{l} x_1 \\ x_2 \\ x_3 \end{array} \quad \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

is an encoding of the solution $(3, 10, 0)$. The language of a formula is then defined to be

$$\mathcal{L}(\varphi) = \{lsbf(s) \mid s \in Sol(\varphi)\}$$

Constructing an NFA for the solution space

Given a formula F , we construct an NFA $\text{Aut}(F)$ such that $L(\text{Aut}(F)) = L(F)$.

We can take:

- $\text{Aut}(\text{not } F) = \text{CompNFA}(\text{Aut}(F))$
- $\text{Aut}(F \text{ or } G) = \text{UnionNFA}(\text{Aut}(F), \text{Aut}(G))$
- $\text{Aut}(\text{Exists } x \ F) = \text{Projection}_x(\text{Aut}(F))$

So it remains to define $\text{Aut}(F)$ for an atomic formula F .

All atomic formulas equivalent (same solutions) to atomic formulas of the form

$$\varphi = a_1x_1 + \dots + a_nx_n \leq b = a \cdot x \leq b$$

where the a_i and b can be arbitrary integers (possibly negative).

Consider a candidate solution

	ζ_0	ζ_1	$\dots$	ζ_m
c_1	$\begin{bmatrix} \zeta_{10} \end{bmatrix}$	$\begin{bmatrix} \zeta_{11} \end{bmatrix}$	$\dots$	$\begin{bmatrix} \zeta_{1m} \end{bmatrix}$
c_2	ζ_{20}	ζ_{21}	$\dots$	ζ_{2m}
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
c_n	$\begin{bmatrix} \zeta_{n0} \end{bmatrix}$	$\begin{bmatrix} \zeta_{n1} \end{bmatrix}$	$\dots$	$\begin{bmatrix} \zeta_{nm} \end{bmatrix}$

For every $j \leq m$, let $c^j \in \mathbb{N}^n$ denote the tuple of numbers encoded by the prefix $\zeta_0 \dots \zeta_{j-1}$. For instance, for the encoding $\zeta_0 \zeta_1 \zeta_2$ of the tuple $(0, 4, 7, 3)$ given by

$$\begin{array}{ccccccc}
 & \zeta_0 & \zeta_1 & \zeta_2 & & \zeta_0 & \zeta_1 \\
 0 & \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} & \text{we get} & 0 & \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} \\
 4 & \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} 1 \end{bmatrix} & & 0 & \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} \\
 7 & \begin{bmatrix} 1 \end{bmatrix} & \begin{bmatrix} 1 \end{bmatrix} & \begin{bmatrix} 1 \end{bmatrix} & & 3 & \begin{bmatrix} 1 \end{bmatrix} & \begin{bmatrix} 1 \end{bmatrix} \\
 3 & \begin{bmatrix} 1 \end{bmatrix} & \begin{bmatrix} 1 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} & & 3 & \begin{bmatrix} 1 \end{bmatrix} & \begin{bmatrix} 1 \end{bmatrix}
 \end{array}$$

and so $c^2 = (0, 0, 3, 3)$. Define further $c^0 = (0, 0, 0, 0)$; i.e., before reading anything all components of the tuple are 0.

We construct a DFA for the solution space of φ . The idea is that after reading a prefix $\zeta_0 \dots \zeta_{j-1}$ the automaton should be in the state

$$\left\lfloor \frac{1}{2^j} (b - a \cdot c^j) \right\rfloor \tag{10.1}$$

Initially we have $c^0 = (0, \dots, 0)$, and so the initial state is the number $\frac{1}{2^0}(b - a \cdot c^0) = b$. For the transitions, assume that before and after reading the letter ζ_j the automaton is in the states q and q' , respectively. Then we have

$$q = \left\lfloor \frac{1}{2^j} (b - a \cdot c^j) \right\rfloor \quad \text{and} \quad q' = \left\lfloor \frac{1}{2^{j+1}} (b - a \cdot c^{j+1}) \right\rfloor$$

From the definition of c^{j+1} we get:

$$c^{j+1} = c^j + 2^j \zeta_j$$

Inserting this in the expression for q' , and comparing with q , we obtain the following relation between q and q' :

$$q' = \left\lfloor \frac{1}{2} (q - a \cdot \zeta_j) \right\rfloor$$

So for every state q and every letter $\zeta \in \{0, 1\}^n$ we take $\delta(q, \zeta) := \frac{1}{2}(q - a \cdot \zeta)$.

PAtoDFA(φ)**Input:** PA formula $\varphi = a \cdot x \leq b$ **Output:** DFA $A = (Q, \Sigma, \delta, q_0, F)$ such that $\mathcal{L}(A) = \mathcal{L}(\varphi)$

```

1   $q_0 \leftarrow s_b$ 
2   $W \leftarrow \{s_b\}$ 
3  while  $W \neq \emptyset$  do
4      pick  $s_k$  from  $W$ 
5      add  $s_k$  to  $Q$ 
6      if  $k \geq 0$  then add  $s_k$  to  $F$ 
7      for all  $\zeta \in \{0, 1\}^n$  do
8           $j \leftarrow \left\lfloor \frac{1}{2}(k - a \cdot \zeta) \right\rfloor$ 
9          if  $s_j \notin Q$  then add  $s_j$  to  $W$ 
10     add  $(s_k, \zeta, s_j)$  to  $\delta$ 

```

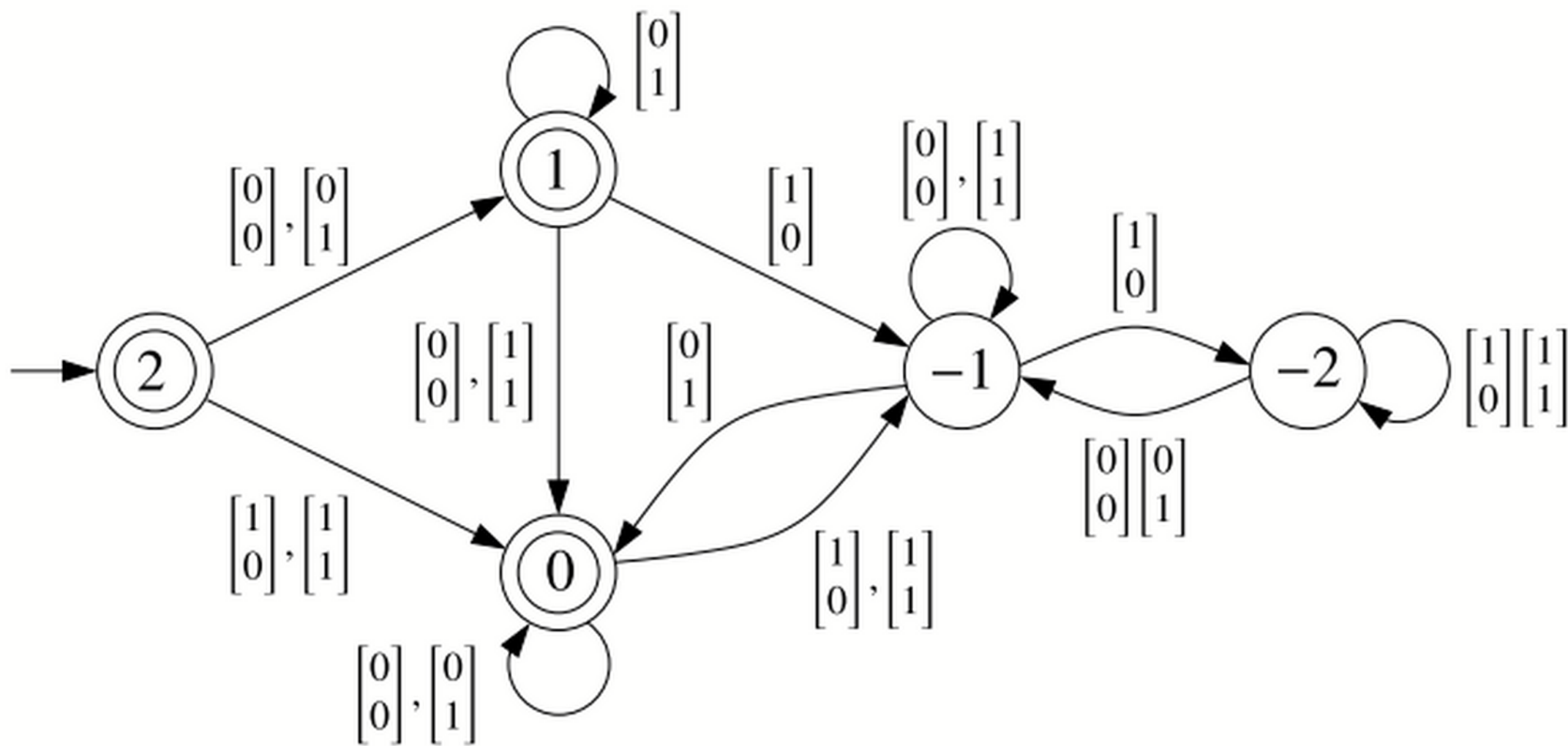


Figure 10.1: DFAs for the formula $2x - y \leq 2$.

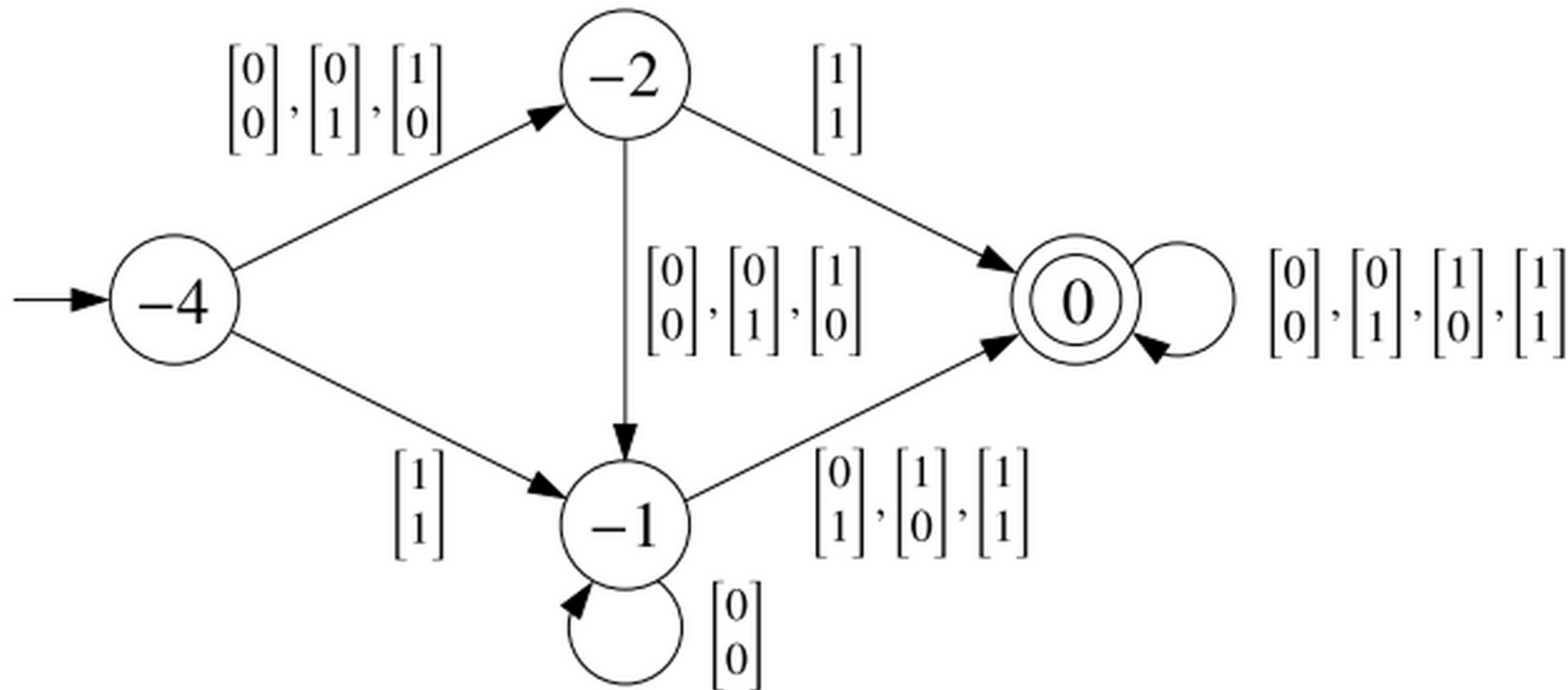


Figure 10.2: DFAs for the formula $x + y \geq 4$.

Lemma 10.3 *Let $\varphi = a \cdot x \leq b$ and $s = \sum_{i=1}^k |a_i|$. All states s_j added to the worklist during the execution of $\text{PAtoDFA}(\varphi)$ satisfy*

$$-|b| - s \leq j \leq |b| + s.$$

Proof: The property holds for s_b , the first state added to the worklist. We show that if all the states added to the worklist so far satisfy the property, then so does the next one.

Let s_j be this next state. Then there exists a state s_k in the worklist and $\zeta \in \{0, 1\}^n$ such that $j = \lfloor \frac{1}{2}(k - a \cdot \zeta) \rfloor$. Since by assumption s_k satisfies the property we have

$$-|b| - s \leq k \leq |b| + s$$

and so

$$\left\lfloor \frac{-|b| - s - a \cdot \zeta}{2} \right\rfloor \leq j \leq \left\lfloor \frac{|b| + s - a \cdot \zeta}{2} \right\rfloor \quad (10.2)$$

Now we observe

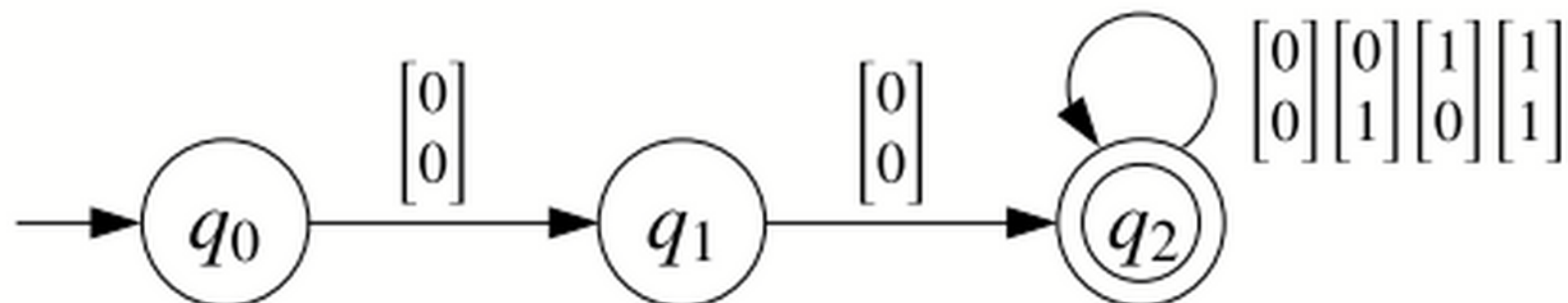
$$\begin{aligned} -|b| - s &\leq \frac{-|b| - 2s}{2} \leq \left\lfloor \frac{-|b| - s - a \cdot \zeta}{2} \right\rfloor \\ \left\lfloor \frac{|b| + s - a \cdot \zeta}{2} \right\rfloor &\leq \frac{|b| + 2s}{2} \leq |b| + s \end{aligned}$$

which together with 10.2 yields

$$-|b| - s \leq j \leq |b| + s$$

and we are done. □

$$\exists z x = 4z \wedge \exists w y = 4w \wedge 2x - y \leq 2 \wedge x + y \geq 4$$



DFA for the formula $\exists z x = 4z \wedge \exists w y = 4w$.

