# Introduction to Formal Methods Chapter 06: Fair CTL Model Checking

#### Roberto Sebastiani and Stefano Tonetta

DISI, Università di Trento, Italy – roberto.sebastiani@unitn.it URL: http://disi.unitn.it/rseba/DIDATTICA/fm2020/ Teaching assistant: Enrico Magnago – enrico.magnago@unitn.it

#### CDLM in Informatica, academic year 2019-2020

last update: Monday 18th May, 2020, 14:48

Copyright notice: some material (text, figures) displayed in these slides is courtesy of R. Alur, M. Benerecetti, A. Cimatti, M. Di Natale, P. Pandya, M. Pistore, M. Roveri, and S. Tonetta, who detain its copyright. Some exampes displayed in these slides are taken from [Clarke, Grunberg & Feled, "Model Checking", MIT Press], and their copyright is detained by the authors. All the other material is copyrighted by Roberto Sebastiani. Every commercial use of this material is strictly forbidden by the copyright laws without the authorization of the authors. No copy of these slides can be displayed in public without containing this copyright notice.

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking





Fair CTL Model Checking: Generalities



Checking Fair EG (Explicit-State)



Checking Fair EG (Symbolic)

#### Outline



#### Fair CTL Model Checking: Generalities







#### Fairness/Justice

An event *E* occurs infinitely often. Example:

- Let  $R_i$  be true iff the process *i* is running.
- Fairness: every process runs infinitely often.

(It is often used as condition for other properties.)

• You cannot express the condition in CTL:

- $\mathbf{GF}\phi = \mathbf{AGAF}\phi$ ,
- but  $(\mathbf{GF}\phi) \rightarrow \psi \neq (\mathbf{AGAF}\phi) \rightarrow \psi$ ,
- because  $FG\phi' \neq EFEG\phi'$ .
- There is no CTL formula equivalent to FG∉ (FGǿ ≠ AFAGø and FGø ≠ EFEGø).

#### Fairness/Justice

An event *E* occurs infinitely often. Example:

- Let *R<sub>i</sub>* be true iff the process *i* is running.
- Fairness: every process runs infinitely often.

(It is often used as condition for other properties.)

• You cannot express the condition in CTL:

- $\mathbf{GF}\phi = \mathbf{AGAF}\phi_i$
- but  $(\mathbf{GF}\phi) \rightarrow \psi \neq (\mathbf{AGAF}\phi) \rightarrow \psi$ ,
- because  $FG\phi' \neq EFEG\phi'$ .
- There is no CTL formula equivalent to  $\mathbf{FG}\phi$
- (FG $\phi \neq$  AFAG $\phi$  and FG $\phi \neq$  EFEG $\phi$ ).

 $(\mathbf{A} \mathbf{GF} R_i)$ 

#### Fairness/Justice

An event *E* occurs infinitely often. Example:

- Let *R<sub>i</sub>* be true iff the process *i* is running.
- Fairness: every process runs infinitely often.

```
\bigwedge_{i} \mathbf{GFR}_{i}
```

(It is often used as condition for other properties.)

- You cannot express the condition in CTL:
  - $\mathbf{GF}\phi = \mathbf{AGAF}\phi$ ,
  - but  $(\mathbf{GF}\phi) \rightarrow \psi \neq (\mathbf{AGAF}\phi) \rightarrow \psi$ ,
  - because  $\mathbf{FG}\phi' \neq \mathbf{EFEG}\phi'$ .
  - There is no CTL formula equivalent to FGφ (FGφ ≠ AFAGφ and FGφ ≠ EFEGφ).

< ロ > < 同 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < 回 > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ > < □ >

- Let *M* be the KS  $M = \langle S, S_0, R, L, AP \rangle$  and  $M_f$  the fair version  $M_f = \langle S, S_0, R, L, AP, FT \rangle$ ,  $FT = \{F_1, ..., F_n\}$ .
- A path is fair iff it visits every *F<sub>i</sub>* infinitely often.
- Fair CTL model checking restricts the path quantifiers to *fair paths*:
  - $M_f, s_i \models \mathbf{A}\phi$  iff  $\pi \models \phi$  for every fair path  $\pi = s_i, s_{i+1}, ...$
  - $M_i, s_i \models \mathbf{E}\phi$  iff  $\pi \models \phi$  for some fair path  $\pi = s_i, s_{i+1}, ...$
- We introduce A<sub>f</sub> and E<sub>f</sub>, the restricted versions of the quantifiers
   A and E:
  - $M_f, s \models \mathbf{A}\phi$  iff  $M, s \models \mathbf{A}_f\phi$
  - $M_f, s \models \mathsf{E}\phi$  iff  $M, s \models \mathsf{E}_f\phi$
- Fair state: a state from which at least one fair path originates, that is, a state *s* is a fair state in  $M_f$  iff  $M_f$ ,  $s \models \mathbf{EG}$  true.

- Let *M* be the KS  $M = \langle S, S_0, R, L, AP \rangle$  and  $M_f$  the fair version  $M_f = \langle S, S_0, R, L, AP, FT \rangle$ ,  $FT = \{F_1, ..., F_n\}$ .
- A path is fair iff it visits every *F<sub>i</sub>* infinitely often.
- Fair CTL model checking restricts the path quantifiers to *fair paths*:
  - $M_i, s_i \models \mathbf{A}\phi$  iff  $\pi \models \phi$  for every fair path  $\pi = s_i, s_{i+1}, \dots$
  - $M_i, s_i \models \mathbf{E}\phi$  iff  $\pi \models \phi$  for some fair path  $\pi = s_i, s_{i+1}, ...$
- We introduce A<sub>f</sub> and E<sub>f</sub>, the restricted versions of the quantifiers
   A and E:
  - $M_f, s \models \mathbf{A}\phi$  iff  $M, s \models \mathbf{A}_f\phi$
  - $M_f, s \models \mathsf{E}\phi$  iff  $M, s \models \mathsf{E}_f\phi$
- Fair state: a state from which at least one fair path originates, that is, a state *s* is a fair state in  $M_f$  iff  $M_f$ ,  $s \models \mathbf{EG}$  true.

- Let *M* be the KS  $M = \langle S, S_0, R, L, AP \rangle$  and  $M_f$  the fair version  $M_f = \langle S, S_0, R, L, AP, FT \rangle$ ,  $FT = \{F_1, ..., F_n\}$ .
- A path is fair iff it visits every *F<sub>i</sub>* infinitely often.
- Fair CTL model checking restricts the path quantifiers to fair paths:
  - $M_i$ ,  $s_i \models \mathbf{A}\phi$  iff  $\pi \models \phi$  for every fair path  $\pi = s_i$ ,  $s_{i+1}$ , ...
  - $M_i$ ,  $s_i \models \mathbf{E}\phi$  iff  $\pi \models \phi$  for some fair path  $\pi = s_i$ ,  $s_{i+1}$ , ...
- We introduce A<sub>f</sub> and E<sub>f</sub>, the restricted versions of the quantifiers
   A and E:
  - $M_f, s \models A\phi$  iff  $M, s \models A_f\phi$
  - $M_f, s \models \mathsf{E}\phi$  iff  $M, s \models \mathsf{E}_f\phi$
- Fair state: a state from which at least one fair path originates, that is, a state *s* is a fair state in  $M_f$  iff  $M_f$ ,  $s \models \mathbf{EG}$  true.

イロト 不得 トイヨト イヨト 二日

- Let *M* be the KS  $M = \langle S, S_0, R, L, AP \rangle$  and  $M_f$  the fair version  $M_f = \langle S, S_0, R, L, AP, FT \rangle$ ,  $FT = \{F_1, ..., F_n\}$ .
- A path is fair iff it visits every *F<sub>i</sub>* infinitely often.
- Fair CTL model checking restricts the path quantifiers to fair paths:
  - $M_i$ ,  $s_i \models \mathbf{A}\phi$  iff  $\pi \models \phi$  for every fair path  $\pi = s_i$ ,  $s_{i+1}$ , ...
  - $M_i$ ,  $s_i \models \mathbf{E}\phi$  iff  $\pi \models \phi$  for some fair path  $\pi = s_i$ ,  $s_{i+1}$ , ...
- We introduce A<sub>f</sub> and E<sub>f</sub>, the restricted versions of the quantifiers A and E:
  - $M_f, s \models A\phi$  iff  $M, s \models A_f\phi$
  - $M_f, s \models E\phi$  iff  $M, s \models E_f\phi$
- Fair state: a state from which at least one fair path originates, that is, a state s is a fair state in M<sub>f</sub> iff M<sub>f</sub>, s ⊨ EGtrue.

- Let *M* be the KS  $M = \langle S, S_0, R, L, AP \rangle$  and  $M_f$  the fair version  $M_f = \langle S, S_0, R, L, AP, FT \rangle$ ,  $FT = \{F_1, ..., F_n\}$ .
- A path is fair iff it visits every *F<sub>i</sub>* infinitely often.
- Fair CTL model checking restricts the path quantifiers to fair paths:
  - $M_i$ ,  $s_i \models \mathbf{A}\phi$  iff  $\pi \models \phi$  for every fair path  $\pi = s_i$ ,  $s_{i+1}$ , ...
  - $M_i$ ,  $s_i \models \mathbf{E}\phi$  iff  $\pi \models \phi$  for some fair path  $\pi = s_i$ ,  $s_{i+1}$ , ...
- We introduce A<sub>f</sub> and E<sub>f</sub>, the restricted versions of the quantifiers A and E:
  - $M_f, s \models A\phi$  iff  $M, s \models A_f \phi$
  - $M_f, s \models E\phi$  iff  $M, s \models E_f\phi$
- Fair state: a state from which at least one fair path originates, that is, a state *s* is a fair state in  $M_f$  iff  $M_f$ ,  $s \models \mathbf{EG}$  true.

- In order to solve the fair CTL model checking problem, we must be able to compute:
  - $[\mathbf{E}_f \mathbf{X}(\varphi)]$
  - [E<sub>f</sub>(φUψ)]
     [E<sub>f</sub>Gφ].
- Suppose we have a procedure Check\_FairEG to compute  $[\mathbf{E}_f \mathbf{G} \varphi]$ .
- Let  $fair = \mathbf{E}_f \mathbf{G} true$ .  $(M, s \models \mathbf{E}_f \mathbf{G} true$  if s is a fair state.)
- if  $\varphi$  is Boolean, then  $M_f, s \models \varphi$  iff  $M, s \models (\varphi \land fair)$
- We can rewrite all the other fair operators:
  - $\mathbf{E}_{f}\mathbf{X}(\varphi) \equiv \mathbf{E}\mathbf{X}(\varphi \wedge fair)$
  - $\mathbf{E}_{f}(\varphi \mathbf{U}\psi) \equiv \mathbf{E}(\varphi \mathbf{U}(\psi \wedge fair))$

- In order to solve the fair CTL model checking problem, we must be able to compute:
  - $[\mathbf{E}_f \mathbf{X}(\varphi)]$
  - $[\mathbf{E}_f(\varphi \mathbf{U}\psi)]$
  - $[\mathbf{E}_f \mathbf{G} \varphi].$

• Suppose we have a procedure Check\_FairEG to compute  $[E_f G \varphi]$ .

- Let fair =  $\mathbf{E}_f \mathbf{G}$ true. ( $M, s \models \mathbf{E}_f \mathbf{G}$ true if s is a fair state.)
- if  $\varphi$  is Boolean, then  $M_f, s \models \varphi$  iff  $M, s \models (\varphi \land fair)$
- We can rewrite all the other fair operators:
  - $\mathbf{E}_f \mathbf{X}(\varphi) \equiv \mathbf{E} \mathbf{X}(\varphi \wedge fair)$
  - $E_f(\varphi U\psi) \equiv E(\varphi U(\psi \wedge fair))$

- In order to solve the fair CTL model checking problem, we must be able to compute:
  - $[\mathbf{E}_f \mathbf{X}(\varphi)]$
  - $[\mathbf{E}_f(\varphi \mathbf{U}\psi)]$
  - $[\mathbf{E}_f \mathbf{G} \varphi].$
- Suppose we have a procedure Check\_FairEG to compute  $[E_f G \varphi]$ .
- Let  $fair = E_f Gtrue$ . ( $M, s \models E_f Gtrue$  if s is a fair state.)
- if  $\varphi$  is Boolean, then  $M_f, s \models \varphi$  iff  $M, s \models (\varphi \land fair)$
- We can rewrite all the other fair operators:
  - $\mathbf{E}_f \mathbf{X}(\varphi) \equiv \mathbf{E} \mathbf{X}(\varphi \wedge fair)$
  - $\mathbf{E}_{f}(\varphi \mathbf{U}\psi) \equiv \mathbf{E}(\varphi \mathbf{U}(\psi \wedge fair))$

- In order to solve the fair CTL model checking problem, we must be able to compute:
  - $[\mathbf{E}_f \mathbf{X}(\varphi)]$
  - $[\mathbf{E}_f(\varphi \mathbf{U}\psi)]$
  - $[\mathbf{E}_f \mathbf{G} \varphi].$
- Suppose we have a procedure Check\_FairEG to compute  $[\mathbf{E}_{f}\mathbf{G}\varphi]$ .
- Let *fair* =  $E_f Gtrue$ . (*M*, *s*  $\models E_f Gtrue$  if s is a fair state.)
- if  $\varphi$  is Boolean, then  $M_f, s \models \varphi$  iff  $M, s \models (\varphi \land fair)$
- We can rewrite all the other fair operators:
  - $\mathbf{E}_f \mathbf{X}(\varphi) \equiv \mathbf{E} \mathbf{X}(\varphi \wedge fair)$
  - $\mathbf{E}_{f}(\varphi \mathbf{U}\psi) \equiv \mathbf{E}(\varphi \mathbf{U}(\psi \wedge fair))$

- In order to solve the fair CTL model checking problem, we must be able to compute:
  - [E<sub>f</sub>X(φ)]
  - $[\mathbf{E}_f(\varphi \mathbf{U}\psi)]$
  - $[\mathbf{E}_f \mathbf{G} \varphi].$
- Suppose we have a procedure Check\_FairEG to compute [E<sub>f</sub>Gφ].
- Let *fair* =  $E_f Gtrue$ . (*M*, *s*  $\models E_f Gtrue$  if s is a fair state.)
- if  $\varphi$  is Boolean, then  $M_f, s \models \varphi$  iff  $M, s \models (\varphi \land fair)$
- We can rewrite all the other fair operators:

• 
$$\mathbf{E}_f \mathbf{X}(\varphi) \equiv \mathbf{E} \mathbf{X}(\varphi \wedge fair)$$

• 
$$\mathbf{E}_{f}(\varphi \mathbf{U}\psi) \equiv \mathbf{E}(\varphi \mathbf{U}(\psi \wedge fair))$$





•  $\mathbf{E}_f(\varphi \mathbf{U}\psi) \equiv \mathbf{E}(\varphi \mathbf{U}(\psi \wedge fair))$ :

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 7/19

э



#### Outline





#### Checking Fair EG (Explicit-State)



# SCC-based Check\_FairEG

A Strongly Connected Component (SCC) of a directed graph is a maximal subgraph s.t. all its nodes are reachable from each other.

Given a fair Kripke model M, a fair non-trivial SCC is an SCC with at least one edge that contains at least one state for every fair condition  $\implies$  all states in a fair (non-trivial) SCC are fair states

# SCC-based Check\_FairEG

A Strongly Connected Component (SCC) of a directed graph is a maximal subgraph s.t. all its nodes are reachable from each other.

Given a fair Kripke model M, a fair non-trivial SCC is an SCC with at least one edge that contains at least one state for every fair condition all states in a fair (non-trivial) SCC are fair states



# SCC-based Check\_FairEG

A Strongly Connected Component (SCC) of a directed graph is a maximal subgraph s.t. all its nodes are reachable from each other.

Given a fair Kripke model M, a fair non-trivial SCC is an SCC with at least one edge that contains at least one state for every fair condition  $\implies$  all states in a fair (non-trivial) SCC are fair states



#### Check\_FairEG( $[\phi]$ ):

- (i) restrict the graph of *M* to  $[\phi]$ ;
- (ii) find all fair non-trivial SCCs  $C_i$
- (iii) build  $C := \cup_i C_i$ ;
- (iv) compute the states that can reach C (Check\_EU ( $[\phi], C$ )).

- Check\_FairEG( $[\phi]$ ):
- (i) restrict the graph of *M* to  $[\phi]$ ;
- (ii) find all fair non-trivial SCCs  $C_i$
- (iii) build  $C := \cup_i C_i$ ;
- (iv) compute the states that can reach C (Check\_EU ([ $\phi$ ], C)).

3

Check\_FairEG( $[\phi]$ ):

- (i) restrict the graph of *M* to  $[\phi]$ ;
- (ii) find all fair non-trivial SCCs C<sub>i</sub>
- (iii) build  $C := \cup_i C_i$ ;

(iv) compute the states that can reach C (Check\_EU ( $[\phi], C$ )).

3

Check\_FairEG( $[\phi]$ ):

- (i) restrict the graph of *M* to  $[\phi]$ ;
- (ii) find all fair non-trivial SCCs C<sub>i</sub>
- (iii) build  $C := \bigcup_i C_i$ ;

(iv) compute the states that can reach C (Check\_EU ( $[\phi], C$ )).

Check\_FairEG( $[\phi]$ ):

- (i) restrict the graph of M to  $[\phi]$ ;
- (ii) find all fair non-trivial SCCs C<sub>i</sub>
- (iii) build  $C := \bigcup_i C_i$ ;
- (iv) compute the states that can reach C (Check\_EU ([ $\phi$ ], C)).

#### $F := \{\{ not C1\}, \{not C2\}\}$ N1, N2 turn=0 N1, T2 T1, N2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1

 $M \models \mathsf{A}_{f}\mathsf{G}(T_{1} \rightarrow \mathsf{A}_{f}\mathsf{F}C_{1}) = \neg \mathsf{E}_{f}\mathsf{F}(T_{1} \land \mathsf{E}_{f}\mathsf{G}\neg C_{1})$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 11/19

э

イロト イポト イモト イモト

#### $F := \{\{ not C1\}, \{not C2\}\}$ N1, N2 turn=0 T1, N2 N1, T2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$

Check\_FairEG( $\neg C_1$ ): 1. compute [ $\neg C_1$ ]

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 11/19

#### $F := \{\{ not C1\}, \{not C2\}\}$ N1, N2 turn=0 T1, N2 N1, T2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$

Check\_FairEG( $\neg C_1$ ): 2. restrict the graph to  $[\neg C_1]$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 11/19

#### $F := \{\{ not C1\}, \{not C2\}\}$ N1. N2 turn=0 T1, N2 N1. T2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=2 turn=2 turn=1 turn=1 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$

Check FairEG( $\neg C_1$ ): 3. find all fair non-trivial SCC's

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

< E Monday 18<sup>th</sup> May, 2020 11/19

**T N** 

#### $F := \{\{ not C1\}, \{not C2\}\}$ N1, N2 turn=0 T1, N2 N1. T2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=2 turn=2 turn=1 turn=1 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$

Check\_FairEG( $\neg C_1$ ): 4. build the union *C* of all SCC's

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

<日 > イヨン イヨン ヨークへの Monday 18<sup>th</sup> May, 2020 11/19

#### $F := \{\{ not C1\}, \{not C2\}\}$ N1. N2 turn=0 T1, N2 N1. T2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=2 turn=2 turn=1 turn=1 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$

Check\_FairEG( $\neg C_1$ ): 5. compute the states which can reach it

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020

11/19

#### $F := \{\{ not C1\}, \{not C2\}\}$ N1, N2 turn=0 N1, T2 T1, N2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1

 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 11/19

э

イロト イポト イモト イモト





 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ Check\_FairEU( $\top, \varphi$ ) = Check\_EU( $\top, \varphi \land fair$ )

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 11/19

э

#### $F := \{\{ not C1\}, \{not C2\}\}$



### $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ fair=Check\_FairEG(T)

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 11/19

э





 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ Check\_FairEU( $\top, \varphi$ ) = Check\_EU( $\top, \varphi \land fair$ )

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 11/19

э

#### $F := \{\{ not C1\}, \{not C2\} \}$ N1, N2 turn=0 N1, T2 T1, N2 turn=2 turn=1 C1. N2 T1, T2 T1. T2 N1, C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1

 $M \models \mathsf{A}_{f}\mathsf{G}(T_{1} \rightarrow \mathsf{A}_{f}\mathsf{F}C_{1}) = \neg \mathsf{E}_{f}\mathsf{F}(T_{1} \land \mathsf{E}_{f}\mathsf{G}\neg C_{1})$ 

э

#### F := {{ not C1},{not C2}}



#### $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ $\implies \text{Property Verified}$

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 11/19

э

- SCCs computation requires a linear (O(#nodes + #edges)) DFS (Tarjan).
- The DFS manipulates the states explicitly, storing information for every state.
- A DFS is not suitable for symbolic model checking where we manipulate sets of states.
- We want an algorithm based on the preimage computation.

- SCCs computation requires a linear (O(#nodes + #edges)) DFS (Tarjan).
- The DFS manipulates the states explicitly, storing information for every state.
- A DFS is not suitable for symbolic model checking where we manipulate sets of states.
- We want an algorithm based on the preimage computation.

- SCCs computation requires a linear (O(#nodes + #edges)) DFS (Tarjan).
- The DFS manipulates the states explicitly, storing information for every state.
- A DFS is not suitable for symbolic model checking where we manipulate sets of states.
- We want an algorithm based on the preimage computation.

- SCCs computation requires a linear (O(#nodes + #edges)) DFS (Tarjan).
- The DFS manipulates the states explicitly, storing information for every state.
- A DFS is not suitable for symbolic model checking where we manipulate sets of states.
- We want an algorithm based on the preimage computation.

#### Outline







#### Checking Fair EG (Symbolic)

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 13/19

Recall the fixpoint characterization of EG:

•  $[\mathbf{EG}\phi] = \nu Z.([\phi] \cap [\mathbf{EX}Z])$ 

The greatest set Z s.t. every state z in Z satisfies  $\phi$  and reaches another state in Z in one step.

We can characterize **E**<sub>f</sub>**G** similarly:

[E<sub>I</sub>Gφ] = νZ.([φ] ∩ ∩<sub>Fi∈FT</sub>[EX E(ZU(Z ∩ F<sub>i</sub>))])
 The greatest set Z s.t. every state z in Z satisfies φ and, for every set F<sub>i</sub> ∈ FT, z reaches a state in F<sub>i</sub> ∩ Z by means of a non-trivial path that lies in Z.



・ 伺 ト ・ ヨ ト ・ ヨ ト

Recall the fixpoint characterization of EG:

•  $[\mathbf{EG}\phi] = \nu Z.([\phi] \cap [\mathbf{EX}Z])$ 

The greatest set Z s.t. every state z in Z satisfies  $\phi$  and reaches another state in Z in one step.

We can characterize  $E_f G$  similarly:

•  $[\mathbf{E}_{f}\mathbf{G}\phi] = \nu Z.([\phi] \cap \bigcap_{F_{i} \in FT} [\mathbf{EX} \mathbf{E}(Z\mathbf{U}(Z \cap F_{i}))])$ The greatest set *Z* s.t. every state *z* in *Z* satisfies  $\phi$  and, for every set  $F_{i} \in FT$ , *z* reaches a state in  $F_{i} \cap Z$  by means of a non-trivial path that lies in *Z*.



```
Recall: [\mathbf{E}_{f}\mathbf{G}\phi] = \nu Z.([\phi] \cap \bigcap_{F_{i} \in FT} [\mathbf{EX} \mathbf{E}(Z\mathbf{U}(Z \cap F_{i}))])
```

```
state_set Check_FairEG( state_set [φ]) {
    Z' := [φ];
    repeat
    Z:= Z';
    for each Fi in FT
        Y:= Check_EU(Z,Fi∩Z);
        Z' := Z' ∩ PreImage(Y));
    end for;
    until (Z' = Z);
    return Z;
```

Implementation of the above formula

}

```
Recall: [\mathbf{E}_{f}\mathbf{G}\phi] = \nu Z.([\phi] \cap \bigcap_{F_{i} \in FT} [\mathbf{EX} \mathbf{E}(Z\mathbf{U}(Z \cap F_{i}))])
```

```
state set Check_FairEG(state set [\phi]) {
     Z' := [\phi];
    repeat
       Z:= Z';
      for each Fi in FT
          Y := Check\_EU(Z', Fi \cap Z');
          Z' := Z' \cap \text{PreImage}(Y));
      end for;
    until (Z' = Z);
    return Z;
}
```

Slight improvement: do not consider states in  $Z \setminus Z'$ 

# Emerson-Lei Algorithm (symbolic version)

```
Recall: [\mathbf{E}_{f}\mathbf{G}\phi] = \nu Z.([\phi] \cap \bigcap_{F_{i} \in FT} [\mathbf{EX} \mathbf{E}(Z\mathbf{U}(Z \wedge F_{i}))])
```

```
Obdd Check_FairEG(Obdd φ) {
   Z':= φ;
   repeat
    Z:= Z';
   for each Fi in FT
    Y:= Check_EU(Z',Fi∧Z');
    Z':= Z' ∧ PreImage(Y));
   end for;
   until (Z' ↔ Z);
   return Z;
```

Symbolic version.

}

#### $F := \{ \{ not C1 \}, \{ not C2 \} \}$



 $M \models \mathsf{AG}(T_1 \rightarrow \mathsf{AF}C_1) = \neg \mathsf{EF}(T_1 \land \mathsf{EG} \neg C_1)$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 18/19

э

# $F := \{ \{ not C1 \}, \{ not C2 \} \}$



 $M \models \mathsf{AG}(T_1 \to \mathsf{AF}C_1) = \neg \mathsf{EF}(T_1 \land \mathsf{EG} \neg C_1)$  $\mathsf{EG}g = \nu Z.g \land \mathsf{EX}Z$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 18/19

э

#### F := { { not C1},{not C2}}



 $M \models \mathsf{AG}(T_1 \to \mathsf{AF}C_1) = \neg \mathsf{EF}(T_1 \land \mathsf{EG} \neg C_1)$  $\mathsf{EG}g = \nu Z.g \land \mathsf{EX}Z$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 18/19

э

#### $F := \{ \{ not C1 \}, \{ not C2 \} \}$ N1, N2 turn=0 N1, T2 T1, N2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models \mathsf{AG}(T_1 \rightarrow \mathsf{AF}C_1) = \neg \mathsf{EF}(T_1 \land \mathsf{EG} \neg C_1)$ $\mathbf{EG}q = \nu Z.q \wedge \mathbf{EX}Z$

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 18/19

э

#### F := { { not C1},{not C2}}



 $M \models \mathsf{AG}(T_1 \to \mathsf{AF}C_1) = \neg \mathsf{EF}(T_1 \land \mathsf{EG} \neg C_1)$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 18/19

э

#### $F := \{ \{ not C1 \}, \{ not C2 \} \}$ N1, N2 turn=0 T1, N2 N1, T2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models AG(T_1 \rightarrow AFC_1) = \neg EF(T_1 \land EG \neg C_1)$

 $\mathbf{EF}g = \mathbf{E}(\top \mathbf{U}g) = \mu Z.g \lor (\top \land \mathbf{EX}Z) = \mu Z.g \lor \mathbf{EX}Z$ < 61 b

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

(2) Monday 18<sup>th</sup> May, 2020 18/19

э

#### $F := \{ \{ not C1 \}, \{ not C2 \} \}$ N1, N2 turn=0 T1, N2 N1, T2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models AG(T_1 \rightarrow AFC_1) = \neg EF(T_1 \land EG \neg C_1)$

 $\mathbf{EF}g = \mathbf{E}(\top \mathbf{U}g) = \mu Z.g \lor (\top \land \mathbf{EX}Z) = \mu Z.g \lor \mathbf{EX}Z$ < 61 b

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

(2) Monday 18<sup>th</sup> May, 2020 18/19

э

#### $F := \{ \{ not C1 \}, \{ not C2 \} \}$ N1, N2 turn=0 T1, N2 N1, T2 turn=1 turn=2 C1. N2 T1. T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models AG(T_1 \rightarrow AFC_1) = \neg EF(T_1 \land EG \neg C_1)$ $\mathsf{EF}g = \mathsf{E}(\top \mathsf{U}g) = \mu Z.g \lor (\top \land \mathsf{EX}Z) = \mu Z.g \lor \mathsf{EX}Z$

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

(4) (2) (4) (3) (4) Monday 18<sup>th</sup> May, 2020 18/19

э

< 6 ×

#### $F := \{ \{ not C1 \}, \{ not C2 \} \}$ N1, N2 turn=0 T1, N2 N1, T2 turn=1 turn=2 C1, N2 T1, T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models AG(T_1 \rightarrow AFC_1) = \neg EF(T_1 \land EG \neg C_1)$ $\mathsf{EF}g = \mathsf{E}(\top \mathsf{U}g) = \mu Z.g \lor (\top \land \mathsf{EX}Z) = \mu Z.g \lor \mathsf{EX}Z$

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 18/19

э

< 6 ×

#### $F := \{ \{ not C1 \}, \{ not C2 \} \}$ N1, N2 turn=0 T1, N2 N1, T2 turn=1 turn=2 C1, N2 T1, T2 T1. T2 N1. C2 turn=1 turn=1 turn=2 turn=2 C1, T2 T1, C2 turn=1 turn=2 not C2 not C1 $M \models AG(T_1 \rightarrow AFC_1) = \neg EF(T_1 \land EG \neg C_1)$

 $\mathsf{EF}g = \mathsf{E}(\top \mathsf{U}g) = \mu Z.g \lor (\top \land \mathsf{EX}Z) = \mu Z.g \lor \mathsf{EX}Z$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 18/19

э

< 6 ×

#### F := { { not C1},{not C2}}



#### $M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1) = \neg \mathbf{EF}(T_1 \land \mathbf{EG} \neg C_1)$ $\implies \text{Property not verified}$

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 18/19

э

#### F := { { not C1},{not C2}}



 $M \models \mathsf{A}_{f}\mathsf{G}(\mathcal{T}_{1} \rightarrow \mathsf{A}_{f}\mathsf{F}\mathcal{C}_{1}) = \neg \mathsf{E}_{f}\mathsf{F}(\mathcal{T}_{1} \land \mathsf{E}_{f}\mathsf{G}\neg \mathcal{C}_{1})$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 19/19

э

イロト イポト イモト イモト

#### F := { { not C1},{not C2}}



 $M \models \mathsf{A}_{f}\mathsf{G}(\mathcal{T}_{1} \rightarrow \mathsf{A}_{f}\mathsf{F}\mathcal{C}_{1}) = \neg \mathsf{E}_{f}\mathsf{F}(\mathcal{T}_{1} \land \mathsf{E}_{f}\mathsf{G}\neg \mathcal{C}_{1})$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 19/19

э

#### F := { { not C1},{not C2}}



 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \wedge \mathbf{E}_{f}\mathbf{G}\neg C_{1})$  $\mathbf{E}_{f}\mathbf{G}g = \nu Z.\underline{g} \wedge \mathbf{EXE}(Z\mathbf{U}(Z \wedge F_{1})) \wedge \mathbf{EXE}(Z\mathbf{U}(Z \wedge F_{2}))$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 19/19

#### F := { { not C1},{not C2}}



# $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \wedge \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ $\mathbf{E}_{f}\mathbf{G}g = \nu Z.g \wedge \mathbf{EXE}(Z\mathbf{U}(Z \wedge F_{1})) \wedge \mathbf{EXE}(Z\mathbf{U}(Z \wedge F_{2}))$

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 1

19/19

#### F := { { not C1},{not C2}}



 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \wedge \mathbf{E}_{f}\mathbf{G}\neg C_{1})$  $\mathbf{E}_{f}\mathbf{G}g = \nu Z.g \wedge \mathbf{E}\mathbf{X}\mathbf{E}(Z\mathbf{U}(Z \wedge F_{1})) \wedge \mathbf{E}\mathbf{X}\mathbf{E}(Z\mathbf{U}(Z \wedge F_{2}))$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020

-

19/19

#### F := { { not C1},{not C2}}



# $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \wedge \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ $\mathbf{E}_{f}\mathbf{G}g = \nu Z.g \wedge \mathbf{EXE}(Z\mathbf{U}(Z \wedge F_{1})) \wedge \mathbf{EXE}(Z\mathbf{U}(Z \wedge F_{2}))$

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 19/19

#### F := { { not C1},{not C2}}



 $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \wedge \mathbf{E}_{f}\mathbf{G}\neg C_{1})$  $\mathbf{E}_{f}\mathbf{G}g = \nu Z.g \wedge \mathbf{EXE}(Z\mathbf{U}(Z \wedge F_{1})) \wedge \mathbf{EXE}(Z\mathbf{U}(Z \wedge F_{2}))$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

✓ □ → < ≥ → < ≥ → </li>
 ■
 Monday 18<sup>th</sup> May, 2020

19/19

#### F := { { not C1},{not C2}}



#### $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ Fixpoint reached

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 19/19

э

#### F := { { not C1},{not C2}}



 $M \models \mathsf{A}_{f}\mathsf{G}(T_{1} \rightarrow \mathsf{A}_{f}\mathsf{F}C_{1}) = \neg \mathsf{E}_{f}\mathsf{F}(T_{1} \land \mathsf{E}_{f}\mathsf{G}\neg C_{1})$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 19/19

э

#### F := { { not C1},{not C2}}



 $M \models \mathsf{A}_{f}\mathsf{G}(T_{1} \rightarrow \mathsf{A}_{f}\mathsf{F}C_{1}) = \neg \mathsf{E}_{f}\mathsf{F}(T_{1} \land \mathsf{E}_{f}\mathsf{G}\neg C_{1})$ 

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 19/19

э

#### F := { { not C1},{not C2}}



#### $M \models \mathbf{A}_{f}\mathbf{G}(T_{1} \rightarrow \mathbf{A}_{f}\mathbf{F}C_{1}) = \neg \mathbf{E}_{f}\mathbf{F}(T_{1} \land \mathbf{E}_{f}\mathbf{G}\neg C_{1})$ $\implies \text{Property verified}$

Sebastiani and Tonetta

Ch. 06: Fair CTL Model Checking

Monday 18<sup>th</sup> May, 2020 19/19

э