# Boolean BI 논리를 위한 컷-제거 귀추 계산법

박종현, 박성우
POSTECH

소프트웨어무결점연구센터 워크샵
2011년 1월 7일

# 연구 개요

1. 목표
   - 포인터 연산 및 동적 메모리 할당을 허용하는 C 프로그램 대상의 연역 검증 도구의 개발

2. 접근 방법
   - Hoare 논리를 확장한 분리 논리 체계를 이용

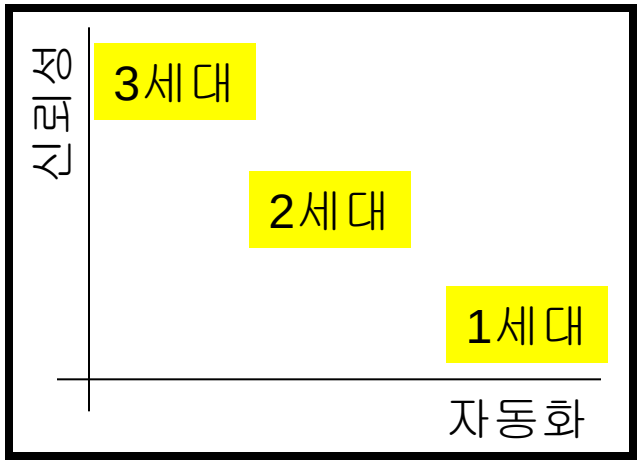
3. 핵심 문제
   - Boolean BI 논리를 위한 컷-제거 귀추계산법 완성

# 1. 목표

포인터 연산 및 동적 메모리 할당을 허용하는
C 프로그램 대상의
연역 검증 도구의 개발

# 프로그램 검증 기술

- 1세대: 테스팅(testing)
  - 테스트 케이스 생성 및 실행
- 2세대: 정적 분석(static analysis)
  - 프로그램 실행 전 소스코드 분석
  - 요약 해석(abstract interpretation), 모델 검증(model checking)



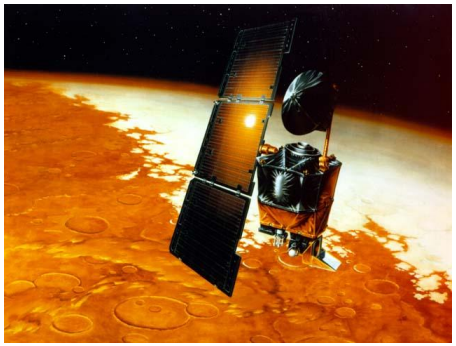

- 3세대: 연역 검증(deductive verification)
  - 정리 증명기(theorem prover), 증명 보조기(proof assistant) 이용
  - 프로그램이 명세를 만족함을 논리 체계를 이용하여 엄밀하게 증명

# 화성 탐사선 Rosaec-Rover

- 운영체제: Rosaec-VxWorks
  - 커널: C 프로그램 1만 라인
- 총 프로젝트 비용: 1조원
  - 발사체, 통신장비, 탐사선, 소프트웨어, ...

*"당신이 프로젝트 책임자라면*
***Rosaec-VxWorks** 커널 검증에 **100**억원을*
*투자하시겠습니까?"*

# Mars Climate Orbiter



- 1998년 NASA에서 발사
- 화성 도착 직후 폭파
- 원인: 부동소수형 변환
  - 모듈 A: English 단위 (feet) 사용
  - 모듈 B: metric 단위 (meter) 사용

# L4.Verified 프로젝트 @ NICTA

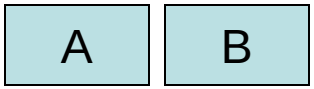
- 목표: ARM11용 seL4 운영체제 마이크로커널 검증
  - C 프로그램 8000 라인
  - Isabelle/HOL 증명 보조기 이용
    - Isabelle 스크립트 20만 라인
- 비용
  - 25 man years
  - $700 per line
- 결과
  - 마이크로커널에 대한 높은 신뢰성

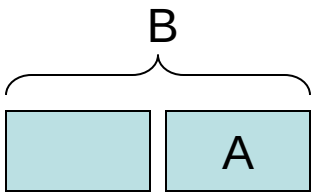
) 결론:
**C 프로그램 연역 검증 도구의 필요성은 더 커질 것이다**

# 2. 접근 방법

Hoare 논리를 확장한 분리 논리 체계를 이용

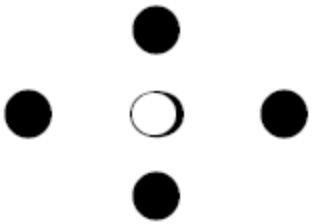
# 분리 논리 (Separation Logic)

- Hoare 논리의 확장 (by John C. Reynolds)
  - 분리 연산자 제공
    - A * B 

    A -* B 

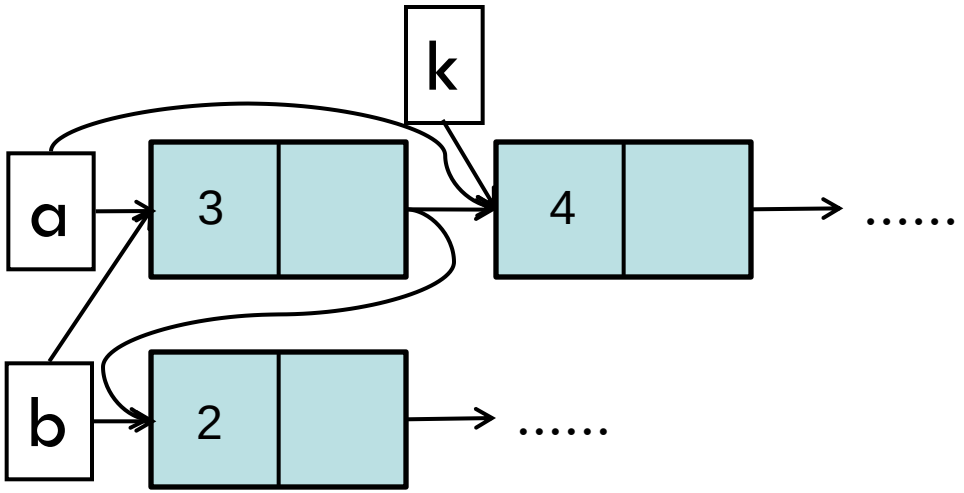
- C 프로그램 연역 검증에 적합
  - 포인터 연산
  - 동적 메모리 할당
  - 가명(alias pointer)

- 지역적 추론(local reasoning)이 가능
  - 사람의 직관적인 추론과 유사



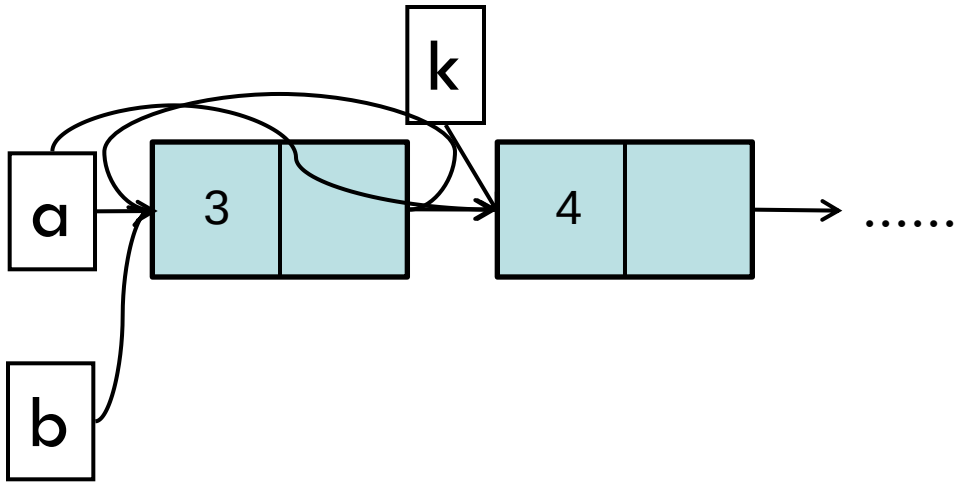
# 리스트 뒤집기

```
b := nil
while a != nil do
  k := [a + 1];
  [a + 1] := b;
  b := a;
  a := k;
end while
```



# 리스트 뒤집기: a = b

```
b := nil
while a != nil do
  k := [a + 1];
  [a + 1] := b;
  b := a;
  a := k;
end while
```




오류 발생???

실제로는 발생하지 않음

# Hoare 논리 vs 분리 논리

```
b := nil
while a != nil do
  k := [a + 1];
  [a + 1] := b;
  b := a;
  a := k;
end while
```

Hoare Logic:

$(\exists\, \alpha, \beta.\ \mathrm{List}\ \alpha\ a \wedge \mathrm{List}\ \beta\ b \wedge {\alpha_0}^R = \alpha^R \cdot \beta) \wedge$
$(\forall\, k.\ \mathrm{Reach}(a, k) \wedge \mathrm{Reach}(b, k) \Rightarrow k = \mathrm{nil})$

Separation Logic:

$(\exists\, \alpha, \beta.\ \mathrm{List}\ \alpha\ a\ *\ \mathrm{List}\ \beta\ b \wedge {\alpha_0}^R = \alpha^R \cdot \beta)$

# 다른 독립된 리스트 x가 있다면...

```
b := nil
while a != nil do
  k := [a + 1];
  [a + 1] := b;
  b := a;
  a := k;
end while
```

Hoare Logic:

$(\exists \alpha, \beta.\ \mathrm{List}\ \alpha\ a \wedge \mathrm{List}\ \beta\ b \wedge {\alpha_0}^R = \alpha^R \cdot \beta)$

$\wedge\ \mathrm{List}\ \gamma\ x$

$(\forall k.\ \mathrm{Reach}(a, k) \wedge \mathrm{Reach}(b, k) \Rightarrow k = \mathrm{nil}) \wedge$

$(\forall k.\ \mathrm{Reach}(x, k) \wedge (\mathrm{Reach}(a, k) \vee \mathrm{Reach}(b, k))$

$\Rightarrow k = \mathrm{nil})$

Separation Logic:

$(\exists \alpha, \beta.\ \mathrm{List}\ \alpha\ a * \mathrm{List}\ \beta\ b * \mathrm{List}\ \gamma\ x \wedge {\alpha_0}^R = \alpha^R \cdot \beta)$

$(\exists \alpha, \beta.\ \mathrm{List}\ \alpha\ a * \mathrm{List}\ \beta\ b \wedge {\alpha_0}^R = \alpha^R \cdot \beta)$

# 3. 핵심 문제

Boolean BI 논리를 위한
컷-제거 귀추계산법 완성

# 연역 검증 과정

소스코드

```
b := nil
while a != nil do
    k := [a + 1];
    [a + 1] := b;
    b := a;
    a := k;
end while
```

⊢ 분리 논리식으로 명세 표현

$(\exists\, \alpha, \beta.\ \mathrm{List}\ \alpha\ a\ *\ \mathrm{List}\ \beta\ b\ \wedge\ \alpha_0{}^{R} = \alpha^{R} \cdot \beta)$

정리증명기/증명보조기

```
(* abs *)
destruct e; simpl in H3; try case_var; inversion H3; subst; simpl; auto.
case_var; try congruence; eauto.
case_var; try congruence.
set (L := FV_ee ({y ::~> Z**A}e)).
destruct (pick_fresh L) as [W Hfresh].
unfold L in Hfresh.
```

**Boolean BI** 증명

$$\dfrac{\dfrac{A \vee B\ true}{(A \vee B) \supset (B \vee A)\ true}\ \supset \mathrm{I}^{x}}{B \vee A\ true}\ \vee \mathrm{E}^{y,z}$$

$$\dfrac{\dfrac{\dfrac{}{A\ true}^{y}}{B \vee A\ true}\ \vee \mathrm{I_R} \qquad \dfrac{\dfrac{}{B\ true}^{z}}{B \vee A\ true}\ \vee \mathrm{I_L}}{B \vee A\ true}\ \vee \mathrm{E}^{y,z}$$

# Boolean BI 논리

- Logic of Boolean Bunched Implications

$$A \quad ::= \quad A \supset A \mid A \vee A \mid A \wedge A \mid \top \mid \bot \mid \neg A \mid A \mathrel{-\!\!\star} A \mid A \star A \mid \mathsf{I}$$

  - 덧셈식 (고전논리) $A \supset A,\ A \vee A,\ A \wedge A,\ \top,\ \bot,\ \neg A$
    - 귀류법 (proof by contradiction) 허용

      $\neg\neg A \supset A$
  - 곱셈식 (BI) $A \mathrel{-\!\!\star} A,\ A \star A,\ \mathsf{I}$
    - 분리 논리에서와 같은 의미

# 핵심 문제: 컷-제거 귀추계산법

1. 귀추계산법 (sequent calculus)
   – Boolean BI를 위한 정리 증명기 구현에 필수적

$$\dfrac{\dfrac{}{A\ true}^{x}\quad\vdots\quad B\ true}{A \supset B\ true}\supset\mathsf{I}^{x} \qquad \dfrac{A \supset B\ true \quad A\ true}{B\ true}\supset\mathsf{E}$$

vs

$$\dfrac{\Gamma, A \supset B \longrightarrow A \quad \Gamma, A \supset B, B \longrightarrow C}{\Gamma, A \supset B \longrightarrow C}\supset L \qquad \dfrac{\Gamma, A \longrightarrow B}{\Gamma \longrightarrow A \supset B}\supset R$$

2. 컷-제거 (cut-free) ¼ Lemma rule

$$\boxed{\text{If } \Gamma \longrightarrow A \text{ and } \Gamma, A \longrightarrow C, \text{ then } \Gamma \longrightarrow C.}$$

   – 설계한 귀추계산법의 건전성 증명에 필수

# 컷-제거 귀추계산법 완성

- 새로운 형태의 귀추(world sequent) 이용

$$
\begin{array}{lrcl}
\text{formula} & A & ::= & P \mid \bot \mid A \wedge A \mid \neg A \mid \mathsf{I} \mid A \mathrel{-\!\!\star} A \mid A \star A \\
\text{boolean bunch} & \Delta & ::= & A \mid \emptyset_{\mathsf{a}} \mid \emptyset_{\mathsf{m}} \mid \Delta ; \Delta \mid W, W \\
\text{falsehood context} & \Psi & ::= & \cdot \mid \Psi ; A \\
\text{world sequent} & W & = & \Delta \longrightarrow_{\mathsf{B}} \Psi
\end{array}
$$

- 컷-제거 증명

$$If\ l_{\mathcal{D}} \sim l_{\mathcal{E}}\ and\ l_{\mathcal{D}}[\Delta \longrightarrow_{\mathsf{B}} \Psi; C]\ and\ l_{\mathcal{E}}[\Delta'; C \longrightarrow_{\mathsf{B}} \Psi'],\ then\ l_{\mathcal{D}} \cdot l_{\mathcal{E}}[\Delta; \Delta' \longrightarrow_{\mathsf{B}} \Psi; \Psi'].$$

$$\frac{A \text{ atomic}}{\omega[A \longrightarrow_{\mathsf{B}} A]}\ \mathit{Init}$$

$$\frac{\omega[\Delta \longrightarrow_{\mathsf{B}} \Psi]}{\omega[\Delta; \Delta' \longrightarrow_{\mathsf{B}} \Psi]}\ \mathsf{W} \qquad \frac{\omega[\Delta \longrightarrow_{\mathsf{B}} \Psi]}{\omega[\Delta \longrightarrow_{\mathsf{B}} \Psi; A]}\ \mathsf{W}' \qquad \frac{\omega[\Delta; \Delta'; \Delta' \longrightarrow_{\mathsf{B}} \Psi]}{\omega[\Delta; \Delta' \longrightarrow_{\mathsf{B}} \Psi]}\ \mathsf{C} \qquad \frac{\omega[\Delta \longrightarrow_{\mathsf{B}} \Psi; A; A]}{\omega[\Delta \longrightarrow_{\mathsf{B}} \Psi; A]}\ \mathsf{C}'$$

$$\frac{}{\omega[\bot \longrightarrow_{\mathsf{B}} \cdot]}\ \bot L \qquad \frac{\omega[\Delta \longrightarrow_{\mathsf{B}} \Psi]}{\omega[\Delta \longrightarrow_{\mathsf{B}} \Psi; \bot]}\ \bot R \qquad \frac{\omega[\Delta \longrightarrow_{\mathsf{B}} A; \Psi]}{\omega[\Delta; \neg A \longrightarrow_{\mathsf{B}} \Psi]}\ \neg L \qquad \frac{\omega[\Delta; A \longrightarrow_{\mathsf{B}} \Psi]}{\omega[\Delta \longrightarrow_{\mathsf{B}} \neg A; \Psi]}\ \neg R$$

$$\frac{\omega[\Delta; A; B \longrightarrow_{\mathsf{B}} \Psi]}{\omega[\Delta; A \wedge B \longrightarrow_{\mathsf{B}} \Psi]}\ \wedge L \qquad \frac{\omega[\Delta \longrightarrow_{\mathsf{B}} A; \Psi] \quad \omega[\Delta \longrightarrow_{\mathsf{B}} B; \Psi']}{\omega[\Delta \longrightarrow_{\mathsf{B}} A \wedge B; \Psi; \Psi']}\ \wedge R$$

$$\frac{\omega[\Delta; \emptyset_{\mathsf{m}} \longrightarrow_{\mathsf{B}} \Psi]}{\omega[\Delta; \mathsf{I} \longrightarrow_{\mathsf{B}} \Psi]}\ \mathsf{I}L \qquad \frac{}{\omega[\emptyset_{\mathsf{m}} \longrightarrow_{\mathsf{B}} \mathsf{I}]}\ \mathsf{I}R$$

$$\frac{\omega[(\Delta' \longrightarrow_{\mathsf{B}} \Psi'; A), (\Delta \longrightarrow_{\mathsf{B}} \Psi); \Delta'' \longrightarrow_{\mathsf{B}} \Psi''] \quad \omega[B; \Delta'' \longrightarrow_{\mathsf{B}} \Psi'']}{\omega[(\Delta' \longrightarrow_{\mathsf{B}} \Psi'), (\Delta; A \mathrel{-\!\!\star} B \longrightarrow_{\mathsf{B}} \Psi); \Delta'' \longrightarrow_{\mathsf{B}} \Psi'']}\ \mathrel{-\!\!\star} L$$

$$\frac{(\Delta \longrightarrow_{\mathsf{B}} \Psi), (A \longrightarrow_{\mathsf{B}} \cdot) \longrightarrow_{\mathsf{B}} B}{\omega[(\Delta \longrightarrow_{\mathsf{B}} A \mathrel{-\!\!\star} B; \Psi), (\Delta' \longrightarrow_{\mathsf{B}} \Psi'); \Delta'' \longrightarrow_{\mathsf{B}} \Psi'']}\ \mathrel{-\!\!\star} R$$

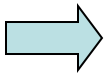
$$\frac{\omega[\Delta; (A \longrightarrow_{\mathsf{B}} \cdot), (B \longrightarrow_{\mathsf{B}} \cdot) \longrightarrow_{\mathsf{B}} \Psi]}{\omega[\Delta; A \star B \longrightarrow_{\mathsf{B}} \Psi]}\ \star L$$

$$\frac{\omega[\Delta''; (\Delta \longrightarrow_{\mathsf{B}} \Psi; A), (\Delta' \longrightarrow_{\mathsf{B}} \Psi') \longrightarrow_{\mathsf{B}} \Psi''] \quad \omega[\Delta''; (\Delta \longrightarrow_{\mathsf{B}} \Psi), (\Delta' \longrightarrow_{\mathsf{B}} \Psi'; B) \longrightarrow_{\mathsf{B}} \Psi'']}{\omega[\Delta''; (\Delta \longrightarrow_{\mathsf{B}} \Psi), (\Delta' \longrightarrow_{\mathsf{B}} \Psi') \longrightarrow_{\mathsf{B}} A \star B; \Psi'']}\ \star R$$

# 비결합적 Classical BI 논리

- 비결합적(non-associative) 분리 연산자

$$A \star (B \star C) \supset (A \star B) \star C$$
$$(A \star B) \star C \supset A \star (B \star C)$$
$$A \star \mathsf{I} \supset A$$
$$A \supset A \star \mathsf{I}$$



증명되지 않음

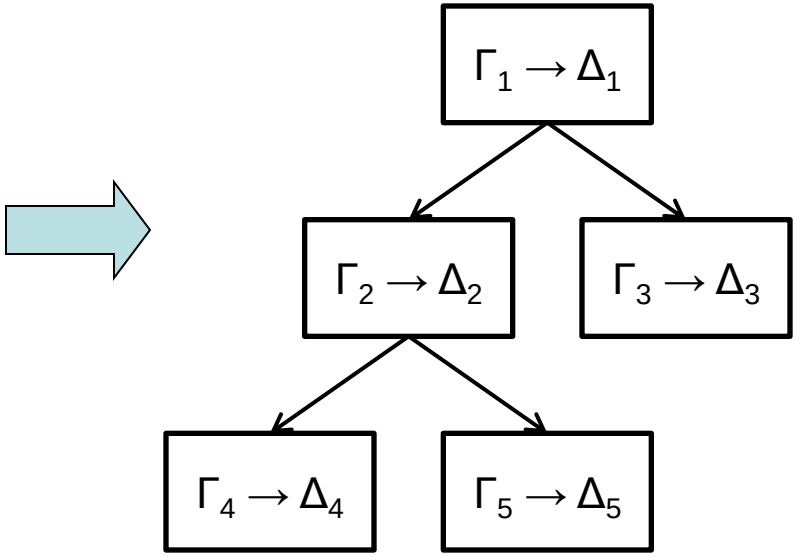
- 곱셈식의 고전적(classical) 해석
  - cf. 직관적(intuitionistic) 해석
- 결론:
  - Boolean BI와는 다른 논리
  - 분리 논리에 이용할 수 없음
  - Boolean BI 논리 컷-제거 귀추계산법은 없음 (99%)
    - Cf. Display calculus를 통한 컷-제거 Boolean BI 정의 [Brotherston 10]

그날 그 이후...

# 비결합적 Classical BI 논리 분석

- 나무 구조의 자원에 대해서만 적용 가능

$$
\begin{array}{lcl}
W & = & \Delta \longrightarrow_{\mathsf{B}} \Psi \\
\Delta & ::= & A \mid \emptyset_{\mathsf{a}} \mid \emptyset_{\mathsf{m}} \mid W, W \mid \Delta; \Delta \\
\Psi & ::= & \cdot \mid A \mid \Psi; \Psi
\end{array}
$$

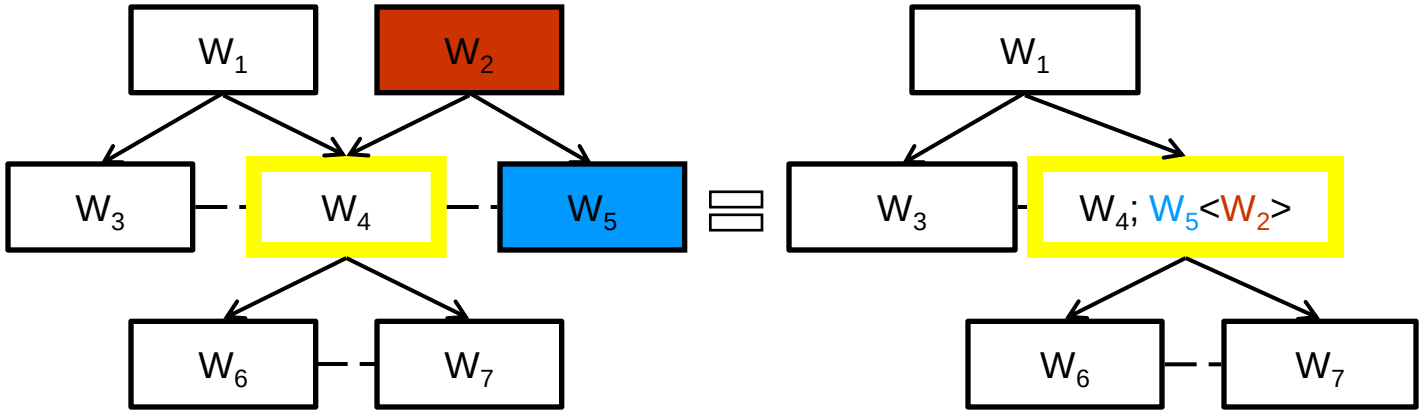



- 선형 구조의 메모리를 나무 구조로 해석해야 함

# 귀추 정의 확장

- “이웃 자원”에 따른 “부모 자원”에 대한 가정 허용

$$
\begin{array}{lcl}
W & ::= & \Gamma \longrightarrow_{\mathsf{B}} \Delta \\
\Gamma & ::= & A \mid \emptyset_{\mathsf{a}} \mid \emptyset_{\mathsf{m}} \mid \Gamma;\Gamma \mid W, W \mid W\langle W\rangle \\
\Delta & ::= & \cdot \mid \Delta; A
\end{array}
$$




- Display calculus를 통한 Boolean BI 정의와 유사

# Boolean BI 컷-제거 귀추 계산법 완성

- 컷-제거 증명

$$If\ \omega[\Gamma \longrightarrow_{\mathsf{B}} \Delta; C]\ and\ \omega'[\Gamma'; C \longrightarrow_{\mathsf{B}} \Delta'],\ then\ \Gamma; \Gamma'; \Gamma_{\omega}; \Gamma_{\omega'} \longrightarrow_{\mathsf{B}} \Delta; \Delta'$$

- 비결합적 classical BI에 비해서
  - 정의가 더 간단
  - 컷-제거 증명이 더 간단
- 현재 진행 중
  - 컷-제거 증명 확인
  - Boolean BI 논리와의 관계 확인
  - 양 20% vs 질 80%

$$\frac{}{P \longrightarrow_{\mathsf{B}} P}\ Init \qquad \frac{\Gamma \longrightarrow_{\mathsf{B}} \Delta}{\Gamma; A \longrightarrow_{\mathsf{B}} \Delta}\ WL \qquad \frac{\Gamma \longrightarrow_{\mathsf{B}} \Delta}{\Gamma \longrightarrow_{\mathsf{B}} \Delta; A}\ WR \qquad \frac{\Gamma; A; A \longrightarrow_{\mathsf{B}} \Delta}{\Gamma; A \longrightarrow_{\mathsf{B}} \Delta}\ CL \qquad \frac{\Gamma \longrightarrow_{\mathsf{B}} \Delta; A; A}{\Gamma \longrightarrow_{\mathsf{B}} \Delta; A}\ CR$$

$$\frac{\Gamma; \emptyset_{\mathsf{a}} \longrightarrow_{\mathsf{B}} \Delta}{\Gamma; \top \longrightarrow_{\mathsf{B}} \Delta}\ \top L \qquad \frac{}{\emptyset_{\mathsf{a}} \longrightarrow_{\mathsf{B}} \top}\ \top R \qquad \frac{\Gamma \longrightarrow_{\mathsf{B}} \Delta; A}{\Gamma; \neg A \longrightarrow_{\mathsf{B}} \Delta}\ \neg L \qquad \frac{\Gamma; A \longrightarrow_{\mathsf{B}} \Delta}{\Gamma \longrightarrow_{\mathsf{B}} \Delta; \neg A}\ \neg R$$

$$\frac{\Gamma; A; B \longrightarrow_{\mathsf{B}} \Delta}{\Gamma; A \wedge B \longrightarrow_{\mathsf{B}} \Delta}\ \wedge L \qquad \frac{\Gamma \longrightarrow_{\mathsf{B}} \Delta; A \quad \Gamma' \longrightarrow_{\mathsf{B}} \Delta'; B}{\Gamma; \Gamma' \longrightarrow_{\mathsf{B}} \Delta; \Delta'; A \wedge B}\ \wedge R$$

$$\frac{\Gamma; \emptyset_{\mathsf{m}} \longrightarrow_{\mathsf{B}} \Delta}{\Gamma; \mathsf{I} \longrightarrow_{\mathsf{B}} \Delta}\ \mathsf{I} L \qquad \frac{}{\emptyset_{\mathsf{m}} \longrightarrow_{\mathsf{B}} \mathsf{I}}\ \mathsf{I} R$$

$$\frac{\Gamma; (A \longrightarrow_{\mathsf{B}} \cdot), (B \longrightarrow_{\mathsf{B}} \cdot) \longrightarrow_{\mathsf{B}} \Delta}{\Gamma; A \star B \longrightarrow_{\mathsf{B}} \Delta}\ \star L \qquad \frac{\Gamma' \longrightarrow_{\mathsf{B}} \Delta'; A \quad \Gamma'' \longrightarrow_{\mathsf{B}} \Delta''; B}{(\Gamma' \longrightarrow_{\mathsf{B}} \Delta'), (\Gamma'' \longrightarrow_{\mathsf{B}} \Delta'') \longrightarrow_{\mathsf{B}} A \star B}\ \star R$$

$$\frac{\Gamma' \longrightarrow_{\mathsf{B}} \Delta'; A \quad \Gamma''; B \longrightarrow_{\mathsf{B}} \Delta''}{(\Gamma' \longrightarrow_{\mathsf{B}} \Delta')\langle \Gamma'' \longrightarrow_{\mathsf{B}} \Delta'' \rangle; A \mathrel{-\!\!\star} B \longrightarrow_{\mathsf{B}} \cdot}\ \mathrel{-\!\!\star} L \qquad \frac{\Gamma; (A \longrightarrow_{\mathsf{B}} \cdot)\langle \emptyset_{\mathsf{a}} \longrightarrow_{\mathsf{B}} B \rangle \longrightarrow_{\mathsf{B}} \Delta}{\Gamma \longrightarrow_{\mathsf{B}} \Delta; A \mathrel{-\!\!\star} B}\ \mathrel{-\!\!\star} R$$

$$\frac{\Gamma''; (\Gamma \longrightarrow_{\mathsf{B}} \Delta), (\Gamma' \longrightarrow_{\mathsf{B}} \Delta') \longrightarrow_{\mathsf{B}} \Delta''}{\Gamma; (\Gamma' \longrightarrow_{\mathsf{B}} \Delta')\langle \Gamma'' \longrightarrow_{\mathsf{B}} \Delta'' \rangle \longrightarrow_{\mathsf{B}} \Delta}\ \Uparrow \qquad \frac{\Gamma; (\Gamma' \longrightarrow_{\mathsf{B}} \Delta')\langle \Gamma'' \longrightarrow_{\mathsf{B}} \Delta'' \rangle \longrightarrow_{\mathsf{B}} \Delta}{\Gamma''; (\Gamma \longrightarrow_{\mathsf{B}} \Delta), (\Gamma' \longrightarrow_{\mathsf{B}} \Delta') \longrightarrow_{\mathsf{B}} \Delta''}\ \Downarrow$$

감사합니다

gla@postech.ac.kr