5. Cálculo de Construcciones Inductivas II - Inversión

Familias y Subfamilias

```
Inductive Even : nat → Prop :=
   e0 : Even 0
|  eSS : forall n:nat, Even n → Even (S (S n)).
```

Even es una familia proposicional indexada por nat

Subfamilias: instancias de familias

- Even 0
- Even (S (S 0))
- Even (S x)
- Even (S 0)

Pruebas sobre familias y subfamilias

```
Inductive Even : nat → Prop :=
   e0 : Even 0
|  eSS : forall n:nat, Even n → Even (S (S n)).
```

Ejemplos:

```
e0                    :  Even 0
eSS 0 e0              :  Even (S (S 0)
eSS (S(S 0)) (eSS 0 e0)     :  Even (S (S (S (S 0))
?????                 : Even (S 0)
```

Esta proposición es Falsa!

Cómo se usa la información sobre subfamilias?

Inductive Even : nat → Prop :=

e0 : Even O

| eSS : forall n:nat, Even n → Even (S (S n)).

Como Even es el conjunto de todos los elementos que se obtienen por aplicación **finita** de los constructores, se demuestra que :

$$\text{Even}(x) \leftrightarrow x=0 \vee \exists y\,(x=(S\,(S\,y)) \wedge \text{Even}(y))$$

Cómo se usa la información sobre subfamilias?

A partir de

$$\text{Even}(x) \leftrightarrow x=0 \ \vee \exists y\,(x=(S\,(S\,y)) \wedge \text{Even}(y))$$

Se puede probar, por ejemplo:

- $\text{Even}(S\ 0) \rightarrow \text{False}$
- $\text{Even}(S\ x) \rightarrow \exists y\,(x=(S\ y) \wedge \text{Even}(y))$
- etc

Generación automática de estas pruebas

Inductive I : $(x_1:V_1)\ (x_2:V_2)\ \ldots(x_n:V_n)$ s:=

c1 : $(y_1:T^1_1)\ldots(y_{n1}:T^1_{n1})\ (I\ t^1_1\ \ldots\ t^1_n)$

| c2 : $(y_1:T^2_1)\ldots(y_{n2}:T^2_{n2})\ (I\ t^2_1\ \ldots\ t^2_n)$

...

| ck : $(y_1:T^k_1)\ldots(y_{nk}:T^k_{nk})\ (I\ t^k_1\ \ldots\ t^k_n)$

1) Se consideran todos los casos, probando el siguiente resultado :

$$(I\ u_1 \ldots\ u_n) \leftrightarrow$$
$$(\exists y_1:T^1_1)\ldots(\exists\ y_{n1}:T^1_{n1})\ \ (u_1 = t^1_1 \wedge \ldots \wedge u_n = t^1_n)$$
$$\vee\ (\exists y_1:T^2_1)\ldots(\exists\ y_{n1}:T^2_{n1})\ \ (u_1 = t^2_1 \wedge \ldots \wedge u_n = t^2_n)$$
$$\ldots$$
$$\vee\ (\exists y_1:T^k_1)\ldots(\exists\ y_{n1}:T^k_{n1})\ \ (u_1 = t^k_1 \wedge \ldots \wedge u_n = t^k_n)$$

Cómo se prueban los principios de análisis de casos para subfamilias?

2) Se simplifican las igualdades de la fórmula obtenida (iterando mientras sea posible `discriminate` e `injection`)

$$(\exists y_1{:}T^1_1)\ldots(\exists\, y_{n1}{:}T^1_{n1})\ (u_1{=}t^1_1 \wedge \ldots \wedge u_n{=}\, t^1_n)$$
$$\vee\ (\exists y_1{:}T^2_1)\ldots(\exists\, y_{n1}{:}T^2_{n1})\ (u_1{=}t^2_1 \wedge \ldots \wedge u_n{=}\, t^2_n)$$
$$\ldots$$
$$\vee\ (\exists y_1{:}T^k_1)\ldots(\exists\, y_{n1}{:}T^k_{n1})\ (u_1{=}t^k_1 \wedge \ldots \wedge u_n{=}\, t^k_n)$$

3) Los casos restantes son un superconjunto de los correspondientes a la subfamilia.

Pueden haber casos cuya no pertinencia deba demostrarse de forma no trivial.

Principios para subfamilias en Coq - Tácticas de inversión

- Los prinicipios de análisis de casos para subfamilias se conocen también como *principios de inversión* (pues *invierten* los constructores).

En Coq hay tres familias de tácticas de inversión:

1. Las que derivan y aplican en línea esos principios:
   inversion, *inversion_clear*

2. Las que derivan y almacenan en el contexto los principios:
   - *Derive Inversion*, *Derive Inversion_clear*
   - *Derive Dependent Inversion, Derive Dependent Inversion_clear*

3. Las que aplican un principio de inversión disponible en el contexto: - *inversion … using ...*

Pruebas por casos en Coq - Inversión

inversion: genera los casos correspondientes a una subfamilia

$$\frac{\Gamma \quad H: (I\ u)}{(P\ u)}\ \textbf{inversion H}\ \frac{\Gamma \quad H: (I\ u) \quad \Delta_{i1}}{(P\ t_{i1})}\ \ldots\ \frac{\Gamma \quad H: (I\ u) \quad \Delta_{ir}}{(P\ t_{ir})}$$

Tales que la prueba H: (I u) puede construirse con r constructores de I.

Los contextos Δ_{ij} son de la forma $x_1{:}U^j_1 \ldots x_{nj}{:}U^j_{nj}$ donde los tipos U^j_h son los tipos de los argumentos del j-esimo constructor de I o son igualdades necesarias para que la prueba H sea obtenida con el j-ésimo constructor (surge de simplificar $u=t^i_j$)

Pruebas por casos en Coq - Inversión

$$\frac{\Gamma \quad H: (I\ u)}{(P\ u)} \qquad \textbf{inversion\_clear H} \qquad \frac{\Gamma \quad H: (I\ u) \quad \Delta_{i1}}{(P\ t_{i1})} \quad \ldots \quad \frac{\Gamma \quad H: (I\ u) \quad \Delta_{ir}}{(P\ t_{ir})}$$

inversion_clear: genera los casos correspondientes a una subfamilia como inversion pero elimina las igualdades iniciales de los contextos Δ_{ij}.

Inversión - Ejemplos

Γ
H: Even (S w)

P w

inversion H

Γ
H: Even (S w)
x: nat
H0: (S x)=w
H1: Even x

P (S x)

Γ
H: Even (S w)

P w

inversion_clear H

Γ
x: nat
H1: Even x

P (S x)