Il principale errore commesso nella procedura è stato quello di eseguire tramite codice PL-SQL ciò che poteva essere demandato alla parte dichiarativa. Oltre a complicare la soluzione ciò rende la procedura molto meno efficiente. Tutta la complessità è spostata su due cursori senza parametri di cui il primo modella direttamente i voli diretti mentre il secondo le combinazioni con uno scalo.

```
(vAreoportoDa VOLI.AreoportoDa%TYPE, vAreoportoA VOLI.AreoportoA%TYPE,
vDataOraPartenza date) IS

cursor voliDiretti_cursor() IS
SELECT V.IDVolo, I.DataOraPartenza, I.DataOraArrivo, I.PrezzoAttuale AS
PrezzoTotale, V.AeroportoA
FROM VOLI AS V, ISTANZEVOLI AS I
WHERE V.IDVolo = I.IDVolo
AND I.DataOraPartenza BETWEEN vDataOraPartenza-2 AND vDataOraPartenza-2
AND I.Disponibilità > 0
AND V.AeroportoDA = vAeroportoDa
AND V.AeroportoA = vAeroportoA;


cursor voliScalo_cursor() IS
SELECT V1.IDVolo, I1.DataOraPartenza, I2.DataOraArrivo, I1.PrezzoAttuale+
I2.PrezzoAttuale AS PrezzoTotale, V2.AeroportoA
FROM VOLI AS V1, ISTANZEVOLI AS I1, VOLI AS V2, ISTANZEVOLI AS I2,
WHERE V1.IDVolo = I1.IDVolo
AND V2.IDVolo = I2.IDVolo
AND I1.DataOraPartenza BETWEEN vDataOraPartenza-2 AND vDataOraPartenza-2
AND V1.AeroportoDA = vAeroportoDa
AND I1.Disponibilità > 0
AND V1.AeroportoA = V2.AeroportoDa
AND I1.DataOraArrivo < I2.DataOraPartenza
AND I2.Disponibilità > 0
AND V2.AeroportoA = vAeroportoA;


vr voliDiretti_cursor%ROWTYPE;

vGap number(4,2);
vNumOre number(5,2);
vIndice number(9,2);

BEGIN

OPEN voliDiretti_cursor();
LOOP
  FETCH voliDiretti_cursor INTO vr;
  EXIT WHEN voliDiretti_cursor%NOTFOUND;

  vGap := ABS(vr.DataOraPartenza - vDataOraPartenza)*24;
  vNumOre := (vr.DataOraArrivo - vDataOraPartenza)*24;
  vIndice := vGap * 10 + vNumOre * 100 + vr.PrezzoAttuale;

  INSERT INTO COMBINAZIONI
    VALUES(vr.IDVolo, vr.DataOraPartenza, vr.PrezzoTotale, vNumOre, 0, vIndice);
CLOSE voliDiretti_cursor;

-- Questa parte è uguale alla precedente, cambia solo il cursore

OPEN voliScalo_cursor();
LOOP
  FETCH voliScalo_cursor INTO vr;
  EXIT WHEN voliScalo_cursor%NOTFOUND;

  vGap := ABS(vr.DataOraPartenza - vDataOraPartenza)*24;
```

```
  vNumOre := (vr.DataOraArrivo - vDataOraPartenza)*24;
  vIndice := vGap * 10 + vNumOre * 100 + vr.PrezzoAttuale;

  INSERT INTO COMBINAZIONI
    VALUES(vr.IDVolo, vr.DataOraPartenza, vr.PrezzoTotale, vNumOre, 0, vIndice);
CLOSE voliScalo_cursor;


END;
```

```
explain plan for select S_SUPPKEY,S_NAME,P_NAME,PS_SUPPLYCOST
from PART,PARTSUPP,SUPPLIER
where PS_PARTKEY=P_PARTKEY and PS_SUPPKEY=S_SUPPKEY and PS_SUPPLYCOST> 700 and
P_TYPE='STANDARD POLISHED NICKEL';
@?/RDBMS/ADMIN/UTLXPLS;
Plan Table
Plan Table
--------------------------------------------------------------------------------
| Operation                 |  Name    |  Rows | Bytes|  Cost  | Pstart| Pstop |
--------------------------------------------------------------------------------
| SELECT STATEMENT          |          |     2K|  214K|   5363 |       |       |
|  HASH JOIN                |          |     2K|  214K|   5363 |       |       |
|   NESTED LOOPS            |          |     2K|  152K|   5164 |       |       |
|    TABLE ACCESS FULL      |PART      |     1K|   78K|   1162 |       |       |
|    TABLE ACCESS BY INDEX R|PARTSUPP  |   240K|    3M|      3 |       |       |
|     INDEX RANGE SCAN      |IX_PART_P |   240K|      |      2 |       |       |
|   TABLE ACCESS FULL       |SUPPLIER  |    10K|  292K|     65 |       |       |
```

$NP_P = \lceil 200.000 \times 130 / (4096 \times 0{,}69) \rceil = 9.200$
$NP_S = \lceil 10.000 \times 143 / (4096 \times 0{,}69) \rceil = 506$
$NP_{PS} = \lceil 800.000 \times 142 / (4096 \times 0{,}69) \rceil = 40.195$

Sel(`PS_SUPPLYCOST> 700`) = (1000-700)/(1000-1) = 0.3
Sel(`P_TYPE='STANDARD POLISHED NICKEL'`) = 1/150;

$NL_{IX\_PART\_SUPPLIER} = \lceil (800.000 \times 4 + 200.000 \times 4) / (4096 \times 0{,}69) \rceil = 1.416$
Costo di ogni accesso a PART_SUPP = 2-1 + $\lceil 1.416 / 200.000 \rceil + \lceil 4/800.000 \times 40195 \rceil = 3$
Costo(NestedLoop$_{P+PS}$)= 9.200 + $\lceil 200.000/150 \rceil \times 3$ = 9.200+4.002=**13.202**

$NT_{P+PS} = \lceil 800.000 \times 1/150 \times 0.3 \rceil = 1.600$
$NP_{P+PS} = \lceil 1.600 \times (130+143) / (4096 \times 0{,}69) \rceil = 155$

Costo (HHJ_{P+PS+S})= $3 \times (155+506)$=1.983

Costo totale 13.202+1.983=15.185