

## PROPAGATING ORIENTATION CONSTRAINTS FOR THE HUBBLE SPACE TELESCOPE

Ashim Bose  
bose@stsci.edu  
and  
Andy Gerb  
gerb@stsci.edu

Space Telescope Science Institute  
3700 San Martin Drive  
Baltimore, MD 21218.

### Abstract

An observing program on the Hubble Space Telescope (HST) is described in terms of exposures that are obtained by one or more of the instruments onboard the HST. Many requested exposures might specify orientation requirements and accompanying ranges. Orientation refers to the amount of roll (in degrees) about the line of sight. The ranges give the permissible tolerance (also in degrees). These requirements may be (i) absolute (in relation to the celestial coordinate system), (ii) relative to the nominal roll angle for HST during that exposure, or (iii) relative (in relation to other exposures in the observing program).

The TRANSformation expert system converts proposals for astronomical observations with HST into detailed observing plans. Part of the conversion process involves grouping exposures into higher level structures based on exposure characteristics. Exposures constrained to be at different orientations cannot be grouped together. Because relative orientation requirements cause implicit constraints, orientation constraints have to be propagated. TRANS must also identify any inconsistencies that may exist so they can be corrected. We have designed and implemented an orientation constraint propagator as part of TRANS. The propagator is based on an informal algebra that facilitates the setting up and propagation of the orientation constraints. The constraint propagator generates constraints between directly related exposures, and propagates derived constraints between exposures that are related indirectly. It provides facilities for path-consistency checking, identification of unsatisfiable constraints, and querying of orientation relationships. The system has been successfully operational as part of TRANS for over seven months. The solution has particular significance to space applications in which satellite/telescope pointing and attitude are constrained and relationships exist between multiple configurations.

### 1 Introduction

This paper discusses the use of constraint satisfaction technology to solve the problem of propagating constraints that determine the amount of roll of a spacecraft/telescope about the line of sight. The solution is geared specifically to the needs of the Hubble Space Telescope (HST), but is general enough to be useful for other satellite applications. Section 1 provides introductions to constraint propagation, TRANSformation (the expert system within which this problem is solved), and the problem domain. Section 2 describes the problem in detail and the implications of this problem on the rest of TRANS. Section 3 discusses the solution that has been incorporated within TRANS. Section 4 includes a small orientation constraint problem and an enumeration of the important steps in arriving at a solution. Section 5 deals with implementation and experience in running the system.

## 1.1. Constraint Propagation

A large number of problems in AI and other areas of computer science can be viewed as special cases of the Constraint-Satisfaction Problem (CSP). The basic CSP can be formulated as follows (Kumar 92): We are given a set of variables, a finite and discrete domain for each variable, and a set of constraints. Each constraint is defined over some subset of the original set of variables and limits the combinations of values that the variables in this subset can take. The goal is to find one or more assignments of values to all variables that satisfies all the constraints. We restrict discussion to CSPs in which each constraint is either unary or binary. These are referred to as binary CSPs.<sup>1</sup>

A CSP can be solved using the generate-and-test paradigm. In this paradigm, all combinations of variable instantiations are systematically generated and tested to see if any satisfy all the constraints. Enhancements to this basic paradigm make it more efficient. For instance, in the backtracking paradigm, variables are instantiated sequentially. If an instantiation violates any of the constraints, backtracking is performed to the most recently instantiated variable that still has alternatives available. Although backtracking is strictly better than the generate-and-test-method, its run-time complexity for most non-trivial problems is still exponential. Thrashing, one of the problems with simple back-tracking, can be attributed in many cases to the lack of arc consistency (Mackworth 77). Arc-consistency algorithms have been studied in Mackworth 77, Mackworth and Freuder 85, Mohr and Henderson 86, Han and Lee 88, Chen 91, Bessiere and Cordier 93. In this scheme, constraints between different variables are propagated to derive a simpler problem. In some cases (depending on the problem and the degree of constraint propagation applied), the resulting CSP is so simple that its solution can be found without search. Although any  $n$ -variable CSP can always be solved by achieving  $n$ -consistency, this approach is usually even more expensive than simple backtracking. A lower-degree consistency (that is,  $K$ -consistency for  $K < n$ ) does not eliminate the need for search except for certain kinds of problems. In Jegou 93, the author uses the concepts of width of hypergraphs and hyper- $K$ -consistency to derive a theorem defining a sufficient condition for consistency of general CSPs.

Several CSP algorithms have been designed for space applications. In order to solve the HST scheduling problem, which is a complex task where between ten thousand and thirty thousand astronomical observations must be scheduled each year subject to a great variety of constraints, Minton and Johnston 90 describe a heuristic repair method called Minimizing-Conflicts, that was inspired by a neural network method described in Adorf and Johnston 90. Miller et al. 88, describe temporal constraints in terms of suitability functions. The suitability function framework has been used successfully as a basis for the SPIKE long-range scheduling system for the HST, as discussed by Johnston and Miller 91. Gerb et al. 92, describe a method based on suitability functions to represent temporal relationships, which is used within the TRANSformation expert system for processing HST proposals.

## 1.2. TRANSformation - The Big Picture

The TRANSformation expert system (TRANS) converts *Proposals* for astronomical observations with the HST into detailed observing plans. It encodes expert knowledge to solve problems faced in planning and commanding HST observations to enable their processing by the Science Operations Ground System (SOGS). Among these problems are determining an acceptable order of executing observations, hierarchical grouping of observations to enhance efficiency and schedulability, inserting extra observations when necessary, and providing

---

<sup>1</sup> A CSP with  $n$ -ary constraints can be converted to an equivalent binary CSP (Rossi et al. 89).

parameters for commanding HST instruments (Gerb 91). TRANS is currently an operational system and plays a critical role in the HST ground system.

### 1.3. The Domain - Controlling Spacecraft Roll

An observing program on the HST is described in terms of exposures that are to be obtained by one or more of the instruments that the HST has on board. These exposures have characteristics that are used by TRANS in its decision making. Many requested exposures might specify orientation requirements. Orientation refers to the amount of roll (in degrees) of the spacecraft about the line of sight. Orientation requirements are specified as ranges that incorporate tolerances. These requirements may be (i) absolute (in relation to the celestial coordinate system), (ii) relative to the nominal roll angle for HST during the exposure, or (iii) relative (in relation to other exposures in the observing program). Constraints between exposures arise due to (iii). These have to be propagated and the constraint network made arc-consistent before TRANS makes its orientation-based decisions.

## 2 The Problem - Orientation Constraints

As already pointed out in the previous section, there are three kinds of observation requirements that an observer may specify for an exposure<sup>2</sup>.

Absolute orientation requirements [(i) above] are of the form:

*REQUIREMENT:* A ORIENT 10 +/- 3D

*EXPLANATION:* The HST orientation for exposure A is to be within an interval of +7 and +13 degrees in relation to the celestial coordinate system.

Orientation requirements relative to the nominal [(ii) above] are of the form:

*REQUIREMENT:* X ORIENT 6 +/- 1D FROM NOMINAL

*EXPLANATION:* The orientation for exposure X should be within an interval of +5 and +7 degrees from the nominal orientation of the HST.

Orientation requirements that are relative [(iii) above] can be further sub-divided into the following categories<sup>3</sup>:

*REQUIREMENT:* X SAME AS Y

*EXPLANATION:* An exposure X has to be at the same orientation as Y.

*REQUIREMENT:* A ORIENT 20 +/- 5D FROM B

*EXPLANATION:* A's orientation has to be at least +15 and at most +25 degrees from B.

TRANS organizes exposures into higher level groups for efficiency and schedulability. There are several rules that are used to determine if two exposures can be grouped (or merged) together. One of these states that two exposures that have a non-free orientation constraint between them cannot be merged (a free constraint is one with a slack of 360 degrees - it has no constraining effect). Hence, during the TRANS merging phase, TRANS needs a way to find out about the existence of a direct or indirect (derived) constraint between any two exposures that are

---

<sup>2</sup>The inequalities of orientation angles and ranges arising from these requirements are stated in Appendix 1.

<sup>3</sup>The actual syntax for these requirements, as used in the Proposals is slightly different. In reality, the SAME AS requirement can arise out of several different requirements.

being considered for merging. A direct constraint between two exposures exists due to the ORIENT FROM requirement. An indirect constraint between two exposures might come about through constraint propagation. For example, consider the following requirements:

A ORIENT 20 +/- 5D FROM B ... (1)

B ORIENT -10 +/- 2D FROM C ... (2)

This implies that there is an indirect constraint between A and C since they are both directly constrained to B. The indirect constraint between A and C can be derived from (1) and (2) by summing up the median values and ranges of the constraints between A and B (1), and B and C (2):

A ORIENT 10 +/- 7D FROM C ... (3)

There may be more than one constraint between two exposures. For example, in addition to the indirect constraint between A and C (3), there may also exist a direct constraint of the form:

A ORIENT 15 +/- 5D FROM C ... (4)

Now, the real constraint between A and C will have to be derived from (3) and (4). In general, the real constraints between two exposures can only be ascertained after all direct and indirect constraints between them have been considered.

Also, an observer may specify requirements that are inconsistent with each other. Consider the requirements:

A ORIENT 20 +/- 5D FROM B ... (1)

B ORIENT -10 +/- 2D FROM C ... (2)

A ORIENT -20 +/- 5D FROM C ... (5)

The direct constraint between A and C as a result of (5) is inconsistent with the indirect constraint (3) derived from (1) and (2).

Another form of inconsistency is due to cyclic dependencies. Consider:

X ORIENT 10 +/- 2D FROM Y

Y ORIENT 12 +/- 4D FROM X

It is impossible to satisfy both requirements simultaneously. On the other hand, not all cyclic dependencies are inconsistent. For example:

X ORIENT 10 +/- 2D FROM Y

Y ORIENT -12 +/- 4D FROM X

can be satisfied.

Inconsistencies resulting from faulty orientation requirements have to be flagged and reported. This is so that the faulty requirements in the Proposal can be altered and the Proposal TRANSed to produce correct output.

### **3 The Solution within TRANS**

The Orient Constraint Propagation module within TRANS sets up and propagates constraints between related exposures. A basic algorithm to achieve this would be the following:

```
arcs-to-check <- all directly related exposure pairs
while arcs-to-check {
  changes <- nil
  for-each arc in arcs-to-check
    changes <- old changes + forward and backward induced arcs from arc
  arcs-to-check <- changes}
```

The actual propagation takes place inside the for-each loop as constraints between exposures that are indirectly related through a common exposure are set up. For example, if there is an arc between exposures i and j, and there is another arc between j and k, an arc is now set up between

i and k. If an arc already exists between i and k, then the resulting constraint is the intersection of the old and propagated constraints. Note that the absolute and nominal ranges for each exposure, as propagated through the constraint, are noted as soon as the new constraint is set up.

This is ordinarily an  $O(n^3)$  operation in the number of related exposures. The number of exposures processed by TRANS can be large (over 100) in some cases. Hence, modifications to the basic algorithm are necessary to ensure that processing time is reasonable.<sup>4</sup>

### 3.1. Representing Angle Intervals and their Intersections

Angle ranges occur in all three kinds of orientation requirements. They are expressed in Proposals as a median value accompanied with a tolerance (see Appendix 2). A more convenient representation is that of a pair, with the first part being the lower limit and the second part being twice the tolerance (or upper limit - lower limit).<sup>5</sup> So, the angle interval  $20 \pm 5D$  can be expressed as (15, 10). Interval intersections are easy to compute.

If  $R_1 = (a_1, b_1)$ , and  $R_2 = (a_2, b_2)$ , then

$$R_3 = R_1 \cap R_2 =$$

$(a_3, b_3)$  such that

$$a_3 = \max(a_1, a_2) \text{ and } b_3 = \min(a_1 + b_1, a_2 + b_2) - a_3 \text{ and } b_3 \geq 0,$$

null (an empty solution set) otherwise.

So, if  $R_1 = (15, 10)$ , and  $R_2 = (12, 8)$ , then  $R_1 \cap R_2 = (15, 5)$ .

### 3.2. Representing a Direct Relationship as a Constraint

A direct ORIENT FROM relationship between two exposures is represented as a constraint interval. For example,

A ORIENT  $20 \pm 5D$  FROM B

gives rise to the constraint

$$C_{AB} = (15, 25).$$

When A is constrained to be at a certain angle interval from B, B is also constrained to be at a certain interval from A. This gives rise to an inverse constraint  $C_{BA}$ . This will be defined in Section 3.4.

### 3.3. Propagating Angle intervals Through Constraints

If exposure B has an angle interval  $R_B = (x_B, y_B)$ , and there exists a constraint between exposures A and B,  $C_{AB} = (l, r)$ , then the angle interval for exposure A,  $R_A$ , propagated from  $R_B$  through constraint  $C_{AB}$ ,

$$R_A = \mathcal{P}(C_{AB}, R_B) = (l + x_B, r + y_B),$$

where  $\mathcal{P}(C_{AB}, R_B)$  denotes the propagation of the angle interval from B to A through the connecting constraint  $C_{AB}$ .

<sup>4</sup>Note that setting up and propagating orientation constraints is only a small part of TRANS processing. For the most part however, TRANS runs in  $O(n)$ .

<sup>5</sup>A more obvious representation would be a pair, where the first member is the lower limit, and the second member is the upper limit. However, in this representation, it is impossible to distinguish a range  $x \pm 0$  degrees from  $x \pm 360$  degrees.

Note that  $R_A$  may already have an angle interval of its own.

Then,

$$R_A^{\text{new}} = R_A^{\text{old}} \cap \mathcal{P}(C_{AB}, R_B).$$

### 3.4. Computing Inverse Relationships

If exposure B has an angle interval  $R_B = (x_B, y_B)$ , and there exists a constraint between exposures A and B,  $C_{AB} = (l, r)$ , then the inverse of this constraint  $C_{BA}$ , is defined to be such that if  $C_{AB} = (l, r)$ , then  $C_{BA} = C_{AB}^{-1} = (-l+r, r)$ .

Let  $R_B = (x_B, y_B)$ ,  $C_{AB} = (l, r)$ , then

$$R_A = \mathcal{P}(C_{AB}, R_B) = (x_B + l, y_B + r).$$

Now, let's propagate a range from A to B.

$$C_{BA} = C_{AB}^{-1} = (-l+r, r).$$

$$R_B \text{ (from range propagated from A)} = R_B^{\text{new}} = \mathcal{P}(C_{BA}, R_A) = (x_B - r, y_B + 2r).$$

$$\text{Hence, } R_B^{\text{new}} \neq R_B^{\text{old}}.$$

But,

$$R_B^{\text{new}} \cap R_B^{\text{old}} = (x_B, y_B) \cap (x_B - r, y_B + 2r) = (x_B, y_B).$$

Hence, even though the range propagated over the inverse constraint has more slack, it doesn't really affect the original range.

### 3.5. Deriving Constraints for Indirect Relationships

An indirect relationship exists between two exposures A and C when they are both related to a third exposure B. Given two constraints

$$C_{AB} = (l_1, r_1), \text{ and } C_{BC} = (l_2, r_2),$$

$C_{AC} = (l_3, r_3)$  is defined to be such that

$$l_3 = l_1 + l_2, \text{ and } r_3 = r_1 + r_2.$$

Consider:

A ORIENT 10 +/- 2D FROM B

B ORIENT 12 +/- 2D FROM C.

This gives rise to the constraints

$$C_{AB} = (8, 4), \text{ and } C_{BC} = (10, 4). \text{ } C_{AC} = (18, 8).$$

Note that as constraints are propagated through a network, the slack (or tolerance - the second part of the constraint pair) can only increase or stay the same. Since the slack is with respect to planar angles, it should be noted that the slack on any constraint cannot exceed 360 degrees. Also, once the slack has reached that limit, the constraint becomes a *free* constraint, i.e. it does not have any constraining effect on the angle intervals of the concerned clan.

### 3.6. An Informal Algebra for Orientation Constraint Propagation

We make modifications to the algorithm mentioned above to make constraint propagation more efficient. These modifications will be described in the following paragraphs.

The SAME AS requirement can be exploited to reduce the value of  $n$  not only in the time and space complexity of the algorithm, but also in the hierarchical ordering of exposures. This is because exposures that have the SAME AS requirement are required to be at the same orientation and can be treated as a group called a *clan*. All exposures that belong to a clan are called its *members*. A clan can have one or more members. By definition, there can be no non-free

ORIENT FROM constraint between two members of the same clan. A constraint between two clans is derived from an ORIENT FROM relationship between its members. There may initially be more than one constraint between two clans. Consider the following case:

A SAME AS B

A ORIENT 10 +/- 2D FROM C

C SAME AS D

D ORIENT -12 +/- 3D FROM B

Exposures A and B belong to the same clan (say clan 1), as do C and D (say clan 2). The following constraints exist between clans 1 and 2 due to the ORIENT FROM requirements:

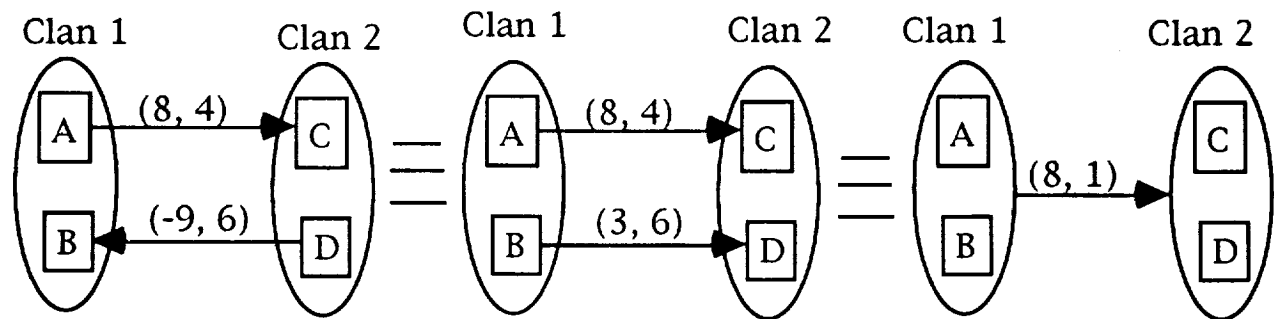
$C_{1A,2C} = (8, 4)$

$C_{2D,1B} = (-9, 6) \Rightarrow C_{1B,2D} = (3, 6)$  (using the inverse relationship formula - Sec. 3.3)

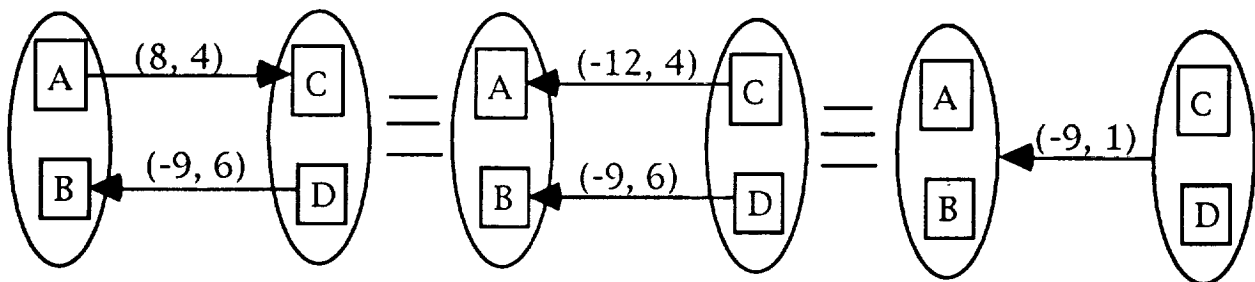
The actual constraints between clans 1 and 2 can be deduced to be:

$C_{12} = C_{1A,2C} \cap C_{1B,2D} = (8, 1)$ , and

$C_{21} = C_{12} = (-9, 1)$ .



Row 1: Steps in generating the constraint  $C_{12}$



Row 2: Steps in generating the constraint  $C_{21}$

**Figure 1: Setting Up and Simplifying Constraints Between Clans Due to Related Member Exposures (either by traversing Row 1 or Row 2).**

In Figure 1, we illustrate how constraints between clans are set up. The constraints can be set up in either direction (by traversing either the first or the second row). Once a constraint between two clans is set up, its inverse yields the constraint in the other direction. Hence there is no need to traverse both rows.

The fact that constraints can be generated between indirectly related clans (via member exposures) helps reduce the propagation time, as the network becomes "stable" faster.

Any interval of the form  $(x, 360)$  where  $x$  is any real number, is a *free* interval. All free intervals are equivalent. Clans have a default interval of  $(0, 360)$  for both absolute and nominal ranges.<sup>6</sup> The absolute and nominal ranges of a clan can change under the following conditions:

- A member exposure has an absolute or nominal range. If  $R_c^{\text{old}}$  is the old range for the clan and  $R_{\text{mem}}$  is the range for the member, the new range for the clan  $R_c^{\text{new}}$  is:

$$R_c^{\text{new}} = R_c^{\text{old}} \cap R_{\text{mem}}.$$

- A range is propagated from a related clan. If  $R_2$  is the range for clan 2 and the constraint between clans 1 and 2 is  $C_{12}$ , then,

$$R_c^{\text{new}} = R_c^{\text{old}} \cap \mathcal{P}(R_2, C_{12}), \text{ where } \mathcal{P}(R, C) \text{ is the result of propagating } R \text{ over } C.$$

If  $R = (x, y)$ , and  $C = (l, r)$ , then  $\mathcal{P}(R, C) = (x + l, y + r)$ .

Consider:

A SAME AS B

A ORIENT 10 +/- 2D FROM C

B ORIENT 20 +/- 5D

C ORIENT 8 +/- 2D

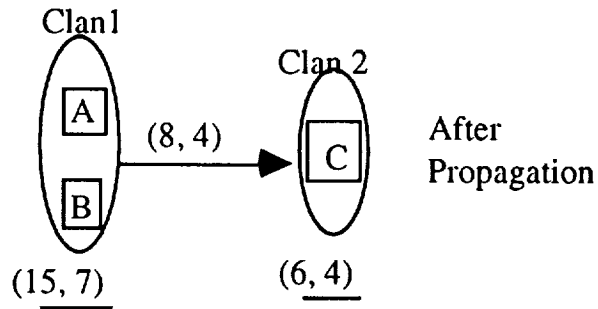
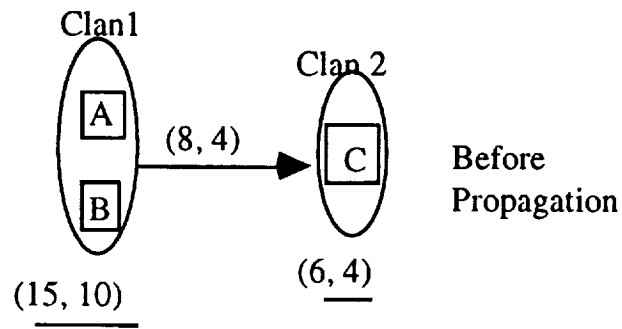
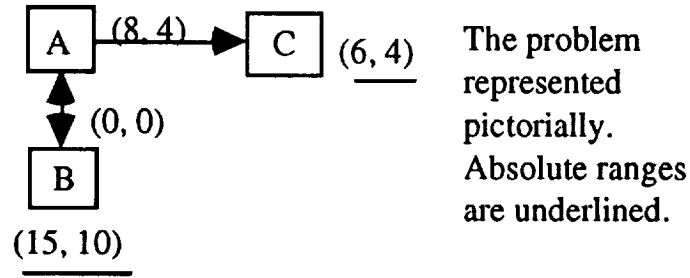
From the above requirements, we have:

Clan 1 = {A, B}, Clan 2 = {C},  $C_{1A,2C} = (8, 4)$ .

---

<sup>6</sup>Note that there is a need to keep track of two sets of intervals for each clan - its absolute interval which is obtained directly or indirectly from requirement (i), and its nominal interval which is obtained directly or indirectly from requirement (ii).





**Figure 2: Propagation of an angle interval Between Two Clans that are Related Through Member Exposures.**

Figure 2 illustrates the propagation of an angle interval from Clan 2 to Clan 1 through the constraint  $C_{12}$ .

$$R_1^{\text{initial}} = R_A \cap R_B = (0, 360) \cap (15, 10) = (15, 10). \quad R_2 = (6, 4).$$

$$R_1^{\text{new}} = R_1^{\text{old}} \cap \mathcal{P}(C_{12}, R_2) = (15, 10) \cap (14, 8) = (15, 7).$$

A constraint network is stable when the absolute and nominal angle intervals of all clans are consistent with the constraints connecting them. This happens only when the constraints between clans cease to change.

### 3.7. Interfacing with the Rest of TRANS

As has already been pointed out, the results of propagating orientation constraints are used within TRANS as one of many inputs to decide on hierarchical exposure groupings. These decisions are made through *merging rules*. The merging rule related to orientations of exposures states that two exposures can be merged only when their absolute and nominal angle intervals are identical and they can be at the same orientation.<sup>7</sup> These requirements will hold only when one of the following conditions is met:

- Both exposures belong to the same clan.
- The exposures belong to different clans that have identical absolute and nominal angle intervals, and either there is no constraint, or a free constraint, or a constraint that includes zero in its interval, between them.

Defective orientation requirements in the Proposal result in null (infeasible) angle intervals for one or more clans and/or null constraints between two or more clans. Warning messages are generated when these are detected, other orientation constraints continue to be propagated, and TRANS uses the results of these propagations to make its merging decisions. In this manner, TRANS products that are not dependent on the faulty orientation requirements can still be used, but the warnings alert the TRANS user to correct the faulty requirements and TRANS the Proposal again, to obtain fully correct TRANS products.

#### 4 A Small Example

We will now illustrate some of the important aspects of orient constraint propagation with the help of a small example. Consider the following orientation requirements:

A ORIENT 20 +/- 5D

B SAME AS A, C

B ORIENT 10 +/- 2D FROM D

C ORIENT 9 +/- 2D FROM F

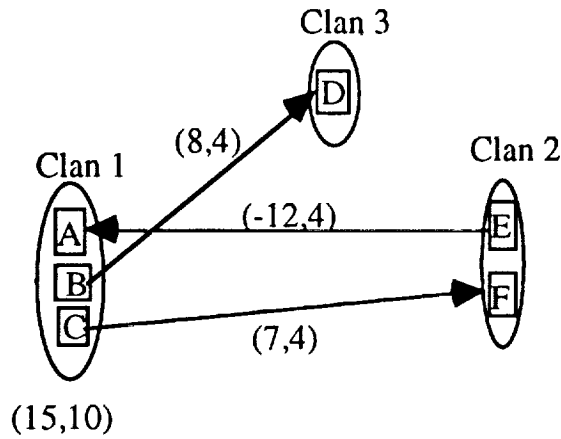
E ORIENT -10 +/- 2D FROM A

E SAME AS F

The problem is represented pictorially in Figure 3a. Clan 1 = {A, B, C}, Clan 2 = {D}, Clan 3 {E, F}.  $R_1 = (15, 10)$ ,  $R_2 = (0, 360)$ ,  $R_3 = (0, 360)$ .  $C_{1B,3D} = (8, 4)$ ,  $C_{1C,2F} = (3, 4)$ ,  $C_{2E,1A} = (-12, 4)$ .

---

<sup>7</sup>Organizing exposures into groups enhances viewing efficiency since the orientation of the spacecraft is the same for all exposures in the group. However, merging exposures that could possibly be at different orientations, into the same group might cause scheduling problems downstream due to the inflexibility that is introduced. Also, the existence of a non-free constraint between two exposures precludes them from being in the same group.



**Figure 3a: Pictorial Representation of the Sample Problem**

Consider the pair Clan 1, Clan 2.

$$C_{21} = C_{2E,1A} \cap C_{1C,2F} = (-12, 4) \cap (-11, 4) = (-11, 3)$$

$$C_{12} = C_{21}^{\sim} = (8, 3)$$

$$R_2 = \mathcal{P}(C_{21}, R_1) = (4, 13)$$

Consider the pair Clan 1, Clan 3

$$C_{13} = (8, 4)$$

$$C_{31} = (-12, 4)$$

$$R_3^{\text{new}} = R_3^{\text{old}} \cap \mathcal{P}(C_{31}, R_1) = (-4, 17) \cap (3, 14) = (3, 10)$$

Now, we propagate constraints.

Deducing the constraint between the pair (Clan 3, Clan 2) from the constraints between the pairs (Clan 3, Clan 1) and (Clan 1, Clan 2),

$$\text{Clan 3} \Leftrightarrow \text{Clan 1} \Leftrightarrow \text{Clan 2}$$

$$C_{32} = (C_{31} + C_{12}) = (-4, 7) \text{ Change}$$

$$C_{23} = (-3, 7)$$

$$R_3^{\text{new}} = R_3^{\text{old}} \cap \mathcal{P}(C_{32}, R_2) = (3, 10) \cap (0, 20) = (3, 10)$$

$$\text{Clan 2} \Leftrightarrow \text{Clan 3} \Leftrightarrow \text{Clan 1}$$

$$C_{21}^{\text{new}} = C_{21}^{\text{old}} \cap (C_{23} + C_{31}) = (-11, 3) \cap (-15, 11) = (-11, 3) - \text{No Change}$$

$$\text{Clan 3} \Leftrightarrow \text{Clan 1} \Leftrightarrow \text{Clan 2}$$

$$C_{32} = (C_{31} + C_{12}) = (-4, 7) \text{ No Change}$$

So, the final solution is:

$$R_1 = (15, 10), R_2 = (4, 13), R_3 = (3, 10)$$

## **5 Implementation and Experience**

The Orient Constraint Propagator, like the rest of TRANS is implemented in LISP, using an object-oriented paradigm (CLOS). Clans are objects, absolute and nominal ranges and constraints are dotted pairs of real numbers. Pairs of related clans, and the constraints between

them, are stored in a hash-table. In order to have a uniform representation for angle intervals, all angles are positive and modulo 360.

The implementation has met all expectations related to run-time efficiency and correctness since it was installed as a TRANS module over seven months ago. It was mentioned in Section 3.5 that the slack in the intervals of constraints can grow to 360 degrees, thus causing propagated constraints to become free. Since in reality, the slack in the intervals of directly related exposures is of the order of ten degrees, this can only happen when there is a long ( $\approx 36$ ) "chain" of related exposures with no interactions with other constraints (note that when there is more than one constraint between two clans, we take their intersection, which has a "constraining effect" on the slack of the resulting constraint). Since the number of exposures in an ORIENT FROM chain is seldom more than five, we have not encountered problems with free constraints.

## **6 Conclusions**

HST observations, expressed as exposures may be related to each other via relationships regulating the roll of the telescope. In this paper, we have formulated these relationships as a CSP, and described a method to "solve" it using an informal algebra. Solving the problem entails determining the ranges of roll for each exposure so that all roll relationships are satisfied. The solution has been implemented using object-oriented technology as a sub-system of the TRANSformation expert system. It has been operational for over seven months and has met all expectations.

## **References**

- Adorf, H. M., and Johnston, M. D. 1990. A Discrete Stochastic Neural Network Algorithm for Constraint Satisfaction Problems. Proceedings of the International Joint Conference on Neural Networks, San Diego, CA.
- Chen, Y. 1991. Improving Han and Lee's Path Consistency Algorithm. In Proceedings of the Third International Conference on Tools for AI, 346-350. Washington, D.C.: IEEE Computer Society.
- Gerb, A. 1991. TRANSformation Reborn: A New Generation Expert System for Planning HST Operations. *Telematics and Informatics* v8n4:283-295.
- Gerb, A., Henry, R., and Johnston, M. 1992. Applying Constraint Propagation Technology to an Expert System Planning Hubble Space Telescope Observations. In Proceedings of the IEEE International Conference on Developing and Managing Intelligent System Projects, 131-138. Washington, D.C.
- Han, C.C., and Lee, C. H. 1988. Comments on Mohr and Henderson's Path-Consistency Algorithm. *Artificial Intelligence* 36: 125-130.
- Johnston, M., and Miller, G. 1991. Long Range Science Scheduling for the Hubble Space Telescope. Proceedings of the 1991 Goddard Conference on Space Applications of Artificial Intelligence, Greenbelt, MD.
- Kumar, V. 1992. Algorithms for Constraint-Satisfaction Problems: A Survey. *AI Magazine* v13(Sp 92):32-44.

Mackworth, A. K. 1977a. Consistency in Networks of Relations. *Artificial Intelligence* 8(1): 99-118.

Mackworth, A. K., and Freuder, E. 1985. The Complexity of Some Polynomial Network-Consistency Algorithms for Constraint-Satisfaction Problems. *Artificial Intelligence* 25:65-74.

Miller, G., Johnston, M., Vick, S., Sponsler, J., and Lindenmayer, K. 1988. Knowledge-based Tools for Hubble Space Telescope Planning and Scheduling, Proceedings of the Goddard Conference on Space Applications of Artificial Intelligence, Greenbelt, MD.

Minton, S., Johnston, M. D., Phillips, A., and Laird, P. 1990. Solving Large-Scale Constraint-Satisfaction and Scheduling Problems Using a Heuristic Repair Method. Proceedings of the Eighth National Conference on Artificial Intelligence, 1990:17-24. Menlo Park, Calif.

Mohr, R., and Henderson, T.C. 1986. Arc and Path Consistency Revisited. *Artificial Intelligence* 28:225-233.

Rossi, F.; Petrie, C.; and Dhar, V. 1989. On the Equivalence of Constraint-Satisfaction Problems, Technical Report ACT-AI-222-89, MCC Corp., Austin, Texas.

## **APPENDIX**

### **(1) ORIENTATION RELATIONSHIPS BETWEEN EXPOSURES:**

In order to determine the merging of exposures based on their orientation requirements, the orientation relationship between pairs of exposures needs to be computed. In the following discussion, we will call the orientation of exposure A Oa. Orientation relationships between exposures come about in six ways:

a. Default - Normally two exposures may be 0d +/- 180d from each other (i.e. they can be at any two orientations).

b. ORIENT FROM - If two exposures A and B are related by A ORIENT <angle> +/- <range> FROM B, the relationship between Oa and Ob should be limited as follows:

$$Ob + \langle \text{angle} \rangle - \langle \text{range} \rangle \leq Oa \leq Ob + \langle \text{angle} \rangle + \langle \text{range} \rangle$$

$$Oa - \langle \text{angle} \rangle - \langle \text{range} \rangle \leq Ob \leq Oa - \langle \text{angle} \rangle + \langle \text{range} \rangle$$

c. SAME ORIENT/SAME POS - If two exposures A and B have SAME POS/ORIENT for A as B. The relationship between Oa and Ob should be limited as follows:

$$Oa = Ob$$

d. Specified orientation - If exposure A has ORIENT <angle1> +/- <range1> and exposure B has ORIENT <angle2> +/- <range2>, or exposure A has ORIENT <angle1> +/- <range1> FROM NOMINAL and exposure B has ORIENT <angle2> +/- <range2> FROM NOMINAL, then the relationship between A and B should be limited as follows:

$$Ob - \langle \text{range2} \rangle - \langle \text{angle2} \rangle + \langle \text{angle1} \rangle - \langle \text{range1} \rangle \leq Oa \leq Ob + \langle \text{range2} \rangle + \langle \text{angle1} \rangle + \langle \text{range1} \rangle - \langle \text{angle2} \rangle$$

$$Oa - \langle \text{range1} \rangle - \langle \text{angle1} \rangle + \langle \text{angle2} \rangle - \langle \text{range2} \rangle \leq Ob \leq Oa + \langle \text{range1} \rangle + \langle \text{angle2} \rangle + \langle \text{range2} \rangle - \langle \text{angle1} \rangle$$

e. Merging - If exposure A is merged into the same scheduling unit (a higher level structure) as exposure B, then the relationship between A and B should be limited as follows:

$$Oa = Ob$$

f. Propagation of constraints - If exposure B is limited by the orientation of A by  $Ob = Oa + \langle \text{angle1} \rangle +/- \langle \text{range1} \rangle$  and exposure C is limited by the orientation of B by  $Oc = Ob + \langle \text{angle2} \rangle +/- \langle \text{range2} \rangle$ , then the relationship between Oa and Oc should be limited as follows:

$$Oa + \langle \text{angle1} \rangle + \langle \text{angle2} \rangle - \langle \text{range1} \rangle - \langle \text{range2} \rangle \leq Oc \leq Oa + \langle \text{angle1} \rangle + \langle \text{angle2} \rangle + \langle \text{range1} \rangle + \langle \text{range2} \rangle$$

$$O_c - \langle \text{angle1} \rangle - \langle \text{angle2} \rangle - \langle \text{range1} \rangle - \langle \text{range2} \rangle \leq O_a \leq O_c - \langle \text{angle1} \rangle - \langle \text{angle2} \rangle + \langle \text{range1} \rangle + \langle \text{range2} \rangle$$

If more than one of the above apply to two exposures A and B, the relationship between  $O_a$  and  $O_b$  should be the intersection of the multiple rules. If this process leads to pairs of exposures A and B where there is no legal angle or range between  $O_a$  and  $O_b$ , TRANS should generate an error.

## (2) ORIENTATION RANGES

Exposures have two sets of limits where they can be scheduled, a nominal range and an absolute range. These ranges are set as follows:

a. An exposure with an ORIENT  $\langle \text{angle} \rangle \pm \langle \text{range} \rangle$  has an absolute range extending from  $\langle \text{angle} \rangle - \langle \text{range} \rangle$  to  $\langle \text{angle} \rangle + \langle \text{range} \rangle$ .

b. An exposure with an ORIENT  $\langle \text{angle} \rangle \pm \langle \text{range} \rangle$  FROM NOMINAL has a nominal range extending from  $\langle \text{angle} \rangle - \langle \text{range} \rangle$  to  $\langle \text{angle} \rangle + \langle \text{range} \rangle$ .

c. An exposure without an ORIENT  $\langle \text{angle} \rangle \pm \langle \text{range} \rangle$  has an absolute range from zero to 360.

d. An exposure without an ORIENT  $\langle \text{angle} \rangle \pm \langle \text{range} \rangle$  FROM NOMINAL has a nominal range from zero to 360.

e. If an exposure B is limited by the orientation of A by  $O_b = O_a + \langle \text{angle1} \rangle \pm \langle \text{range1} \rangle$ , where A has an absolute range from  $\langle \text{low-angle} \rangle$  to  $\langle \text{high-angle} \rangle$  then B has an absolute range from  $\langle \text{low-angle} \rangle + \langle \text{angle1} \rangle - \langle \text{range1} \rangle$  to  $\langle \text{high-angle} \rangle + \langle \text{angle1} \rangle + \langle \text{range1} \rangle$ .

If more than one of the above rules apply to an exposure A, the ranges of A should be the intersection of ranges specified by the multiple rules. If the resulting interval is empty for any given exposure, TRANS should generate an error.