UCRL-16478

# University of California

# Ernest O. Lawrence Radiation Laboratory

MULTIDIMENSIONAL CORRELATION LATTICES AS AN AID TO THREE-DIMENSIONAL PATTERN RECONSTRUCTION

TWO-WEEK LOAN COPY

*This is a Library Circulating Copy which may be borrowed for two weeks. For a personal retention copy, call Tech. Info. Division, Ext. 5545*

Berkeley, California

## DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

For the Spring Joint Computer Conference
April 26-28, 1966, Boston, Mass.

UCRL-16478

UNIVERSITY OF CALIFORNIA

Lawrence Radiation Laboratory
Berkeley, California

AEC Contract No. W-7405-eng-48

# MULTIDIMENSIONAL CORRELATION LATTICES AS AN AID TO THREE-DIMENSIONAL PATTERN RECONSTRUCTION

Samuel J. Penny and James H. Burkhard

January 26, 1966

# MULTIDIMENSIONAL CORRELATION LATTICES AS AN AID TO THREE-DIMENSIONAL PATTERN RECONSTRUCTION

Samuel J. Penny and James H. Burkhard

Lawrence Radiation Laboratory
University of California
Berkeley, California

January 26, 1966

## ABSTRACT

A technique has been developed to perform an efficient and exhaustive search for solutions of certain types of combinatorial problems. The basic tool of the technique is the "correlation lattice" for which we present the properties and algorithms for manipulation. The technique is described as applied to the spatial reconstruction of a nuclear-particle interaction from three stereo photographs. In this example the particle-track images in each of the three stereo views must first be unambiguously correlated and the spurious images discarded. Then the spatial trajectories of the particles may be computed. Though illustrated for the nuclear-particle application, the technique is also applied in other fields.

## INTRODUCTION

Spatial reconstruction of a three-dimensional object from a set of stereo photographs must begin with the matching of the topological characteristics in one view with those in the other views. A human is usually good at this form of pattern recognition, but he is too slow for some applications. For high-speed processing of bubble chamber photographs it has been necessary to design a computer code that will match the images seen in the various views.

If we have n stereo views of an object, where each view contains images of the m characteristics to be correlated, a total of $m^n$ combinations are possible when one image is used from each view. These combinations are known as "n-tuplets". There is then some set containing m of the n-tuplets that describes the true matching of the images. With the physical limitation that an image in one view should not match more than one image in any other view, this solution set must be chosen from a total of $(m!)^{n-1}$ possible sets. This is in the class of combinatorial problems that involve the construction of an ordered set $S = \{ s_1,\ldots,s_m\}$ where the $s_i$ are elements of a finite set U and the elements of the set S must be chosen subject to certain restrictions.

One approach for finding the correct solution set would be to compute the likelihood for each possible set of m n-tuplets, and then choose that set with the maximum likelihood. This approach can require a prohibitive amount of computation and is unreliable when there are large errors in the data. An alternative approach would be to use some type of elimination process, but even this can be expensive unless a technique is used to simplify the bookkeeping and to provide a rapid correlation among the various steps in the process. The "correlation lattice" and methods for scanning it for solution sets (by means of an iterative scheme of backtrack programming[1,2]) supply the simplicity and speed to make this approach economical on a computer.

## THE THREE-DIMENSIONAL LATTICE

At the Lawrence Radiation Laboratory in Berkeley, much of the high-energy-physics research deals with nuclear-particle events occurring in a bubble chamber. Three stereo views of the event are photographed, and points along the track images are measured. The images must then be matched so that the spatial trajectories of the particles involved in the interaction can be mathematically reconstructed. Using the parameters of the trajectories, we do a kinematic analysis of the interaction to produce the information required by the physicist for his experiments.

Figure 1 is a photograph and sketches of a simple nuclear event that occurred in the bubble chamber. As charged nuclear particles move through the liquid in the chamber, they leave small tracks of bubbles. A nuclear event occurs when an elementary particle interacts with an atomic nucleus in the liquid and produces a new group of particles moving in different directions.

Numbers have been assigned to the track images in the different views in the figure. The set of image-number triplets that corresponds to the correct track-image matching is

$$S = \{(1,1,3), (2,3,2), (3,2,1)\} , \qquad (1)$$

where, for example, the triplet (1,1,3) means that track image 1 in view 1, track image 1 in view 2, and track image 3 in view 3 are images of the same physical track.

### PRODUCING THE LATTICE

The computer's procedure for matching the track images begins with the construction of the correlation lattice, E. Each dimension of this lattice corresponds to a set of track images from one of the stereo views, and the indices along a dimension are the identifying numbers of the track images in that view. The value, 1 or 0, of an element $e_{ijk}$ in E will indicate the possibility or impossibility, respectively, of matching image i in view 1 with image j in view 2 with image k in view 3. Since all image combinations must initially be assumed to be possible, the

procedure begins with all lattice elements equal to 1. Figure 2 shows the initial lattice for our sample event.

Often the correlation lattice will not be cubic, as it was for our sample event. It is possible that spurious images may be seen or real images missed in some views. With the bubble chamber film, it may be difficult to dissociate the background tracks from the interesting interaction in some view, thus producing spurious images in that view. However, the measuring devices seldom miss track images. The strategy for dealing with the noncubic lattice for track matching is then rather simple: if L is the number of images the code "expects" to match, then every view must have at least L measured track images, and the solution set must contain L triplets. Other applications may require other strategies.

The remainder of the matching procedure is an iterative process of applying successively more difficult tests of physical consistency to the image combinations in an attempt to eliminate matching possibilities from the correlation lattice, and, after each test sequence, scanning the lattice to determine the number of possible solutions sets that remain. If there is no such set, or only one, the procedure ends. If there is more than one set, the lattice is still ambiguous, and it is necessary to go into the next test sequence.

## TESTING IMAGE COMBINATIONS

A test may be concerned with an image from each of the three views (a triplet test) or it may compare images from only two views (a pair test). The pair test is very effective in the early stages of the procedure since, if an image pair is shown to be inconsistent, any triplet involving that pair must also be inconsistent. A single test can therefore eliminate many elements at a time.

The tests may be expressed as binary-valued functions. A pair test is written as the binary function $F_{P,Q}(p,q)$ of the pair of image indices p and q from views P and Q. When the image pair is not consistent with physical constraints, the value of the function is 0; otherwise its value is 1. A similar definition is made of the triplet test $F_{1,2,3}(i,j,k)$.

For efficiency, we do not apply a test to any image combination that has already been eliminated by a previous test sequence. The determination of triplets to be tested with a triplet test is straightforward: A triplet is tested only if its corresponding element in E has the value of 1. For the set of pair tests ($F_{1,2}$; $F_{2,3}$; $F_{3,1}$), one view at a time must be systematically eliminated from the image combinations by forming the "projections" of E along each of its three dimensions, one at a time. This produces the three two-dimensional-projection lattices ${}_1E$, ${}_2E$, and ${}_3E$, where the elements of these lattices are given by

$$ {}_1E = \left({}_1e_{jk}\right) = \left( \underset{i=1}{\overset{n_1}{.\mathrm{OR}.}} e_{ijk} \right) , \tag{2a} $$

$$ {}_2E = \left({}_2e_{ij}\right) = \left( \underset{j=1}{\overset{n_2}{.\mathrm{OR}.}} e_{ijk} \right) , \tag{2b} $$

$$ {}_3E = \left({}_3e_{ij}\right) = \left( \underset{k=1}{\overset{n_3}{.\mathrm{OR}.}} e_{ijk} \right) . \tag{2c} $$

.OR. is the logical sum operator, i.e.,

| OR | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 1 | 1 |

The image pairs to be tested are determined by scanning the three projection lattices for elements with a value of 1 and testing only the corresponding pairs. The projections of the initial lattice shown in Fig. 2 will obviously produce projection lattices that have elements all equal to 1.

RECORDING TEST RESULTS

Now that the image combinations to be tested are determined, it is now necessary to record the results of the test sequence in the correlation lattice E. For a triplet test this simply involves setting the lattice element corresponding to the triplet to 0 if the value of the test function is 0. The procedure for recording the results of a pair-test sequence is to record the value of the test function in the two-dimensional-projection lattices that indicated the image pairs to be tested. After all image pairs have been tested, the resulting projection lattices are used to "mask" the correlation lattice E. This masking operation is written as

$$E' = {}_kE.\mathrm{AND}.E \quad , \tag{3}$$

or expressed at the element level for the view 2/3 projection as

$$e'_{ijk} = ({}_1e_{jk}).\mathrm{AND}.(e_{ijk}) \tag{4}$$

for all i, j, and k. Here, .AND. is the logical product operator

| AND | 0 | 1 |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |

The masking is done for all three projections, and the final lattice E' replaces E as the correlation lattice.

Returning to our sample event, suppose that the first test sequence was a pair test and eliminated five image pairs: (1,2) and 2,2) were inconsistent in views 1 and 2, and (1,1), (2,1), and (2,3) were inconsistent in views 1 and 3. No pairs could be eliminated from views 2 and 3. Figure 3 shows the projection lattices containing the test results ready to be masked into the correlation lattice E.

SCANNING FOR SOLUTIONS

After masking into the correlation lattice, we now scan the three-dimensional lattice for possible solutions. A solution in the lattice is defined as a set of lattice elements that all have a value of 1 and are chosen in such a way that no image in any view is used more than

once. The length L of the solution set is defined as the number of elements in the solution set; in our sample event, the length is three.

Figure 4 shows the results of the masking operation on our sample event. The dashed lines represent the two possible solution sets that will be found by the scanning procedure.

The search for solutions in the three-dimensional lattice is based on the following proposition: If there exists a solution set consisting of L triplets -- $S_L(E)$ -- in the three-dimensional lattice E, then at least one two-dimensional-solution set consisting of L doublets -- $S_L({}_kE)$ -- must also exist in each of the projection lattices of E; i.e. a solution in E will project as a solution in any projection of E. A corollary to this proposition is that if no solution set of length L is found in some projection of E, then no solution set of length L exists in E.

The scanning procedure is a four-step process. In the first step the lattice is reprojected along some axis to produce ${}_kE$ (${}_3E$ in our case). In the second step this two-dimensional lattice is scanned for two-dimensional solutions of the required length by the technique of backtrack programming. When a solution is found, the index pairs forming the solution set are used to construct a two-dimensional "post-projection" lattice P (step three) as described later. In step four this lattice is then scanned for solutions in order to determine the third index of the solution triplets. This type of scan fixes the indices of the triplets one view at a time--i.e., i is fixed for view 1, then corresponding j's are chosen from view 2, and this information is used to pick out the corresponding k's from view 3. These steps are illustrated in Fig. 5.

As the three-dimensional solutions are found, they are used to construct a new three-dimensional lattice, $E_S$, that contains only elements known to be a part of some solution. This lattice will be used later to discard unused elements on the basis of logical inconsistency. (If it is found that an element is not now a part of some solution in E, it can never be a part of any solution, and it can then therefore be ignored in any further processing.)

The first step--to produce ${}_3E$--is the projection procedure of Eq. (2c). By the proposition stated above, any solution in E will project as a solution in ${}_3E$.

The second step--to scan ${}_3E$ for a two-dimensional solution--is done with backtrack programming[1] in the following way. Suppose that we are dealing with a projection lattice of n rows and m columns and are searching for solutions of length L. Row i of the lattice is used to form the set $R_i = \{r_{ij}\}$, where the first m members of the set have values of 0 or 1, equal to the corresponding elements of the row, and the (m+1)th member of the set is always 0. (This last member represents a dummy track used to flag the row as not being used in a solution set.) Using these sets to form the Cartesian product space $R_1 \times R_2 \times \ldots \times R_n$, we search for a vector $(r_{1j_1}, r_{2j_2}, \ldots, r_{nj_n})$ in the space that satisfies the criterion function

$$f(r) = \sum_{i=1}^{n} r_{ij_i} = L \qquad (5)$$

and that meets the constraint that no two $j_i$ indices can have the same value unless that value is m+1. (This constraint allows any column of the projection lattice to be used only once.)

The concept of backtracking is to construct the vector one component at a time, with modified criterion functions used at each step to determine if the line of pursuit still has a chance of success. When it is found that a partially constructed vector is doomed to failure, the last component is discarded, and the program backtracks to resume construction from the preceding elements.

The modified criterion function used in the scan of a projection lattice at the kth row is

$$f_k(r) = \sum_{i=1}^{k} r_{ij_i} \geq L - (n-k) \qquad (6)$$

i.e., there must be enough rows left for the scan to reach the final value

of L. When n and L are equal (as in our sample event), this function reduces to

$$f_k(r) = k \ . \tag{7}$$

In Fig. 5, the backtrack program begins by selecting $r_{11}$ from set $R_1$ and testing it with the modified criterion function $f_1(r)$. In this case

$$f_1(r_{11}) = 1 \ ,$$

and the criterion is met. Having found the first component of our sample vector, $(r_{11}, -, -)$, the program now begins a search in $R_2$. The first choice is $r_{21}$ and by applying Eq. (7) we see that

$$f_2(r_{11}, r_{21}) = 2 \ .$$

However, the constraint that a column may be used only once eliminates this path from consideration. The component $r_{21}$ is discarded, and we move to $r_{22}$. Hear again we meet failure because

$$f_2(r_{11}, r_{22}) < 2 \ .$$

Continuing, we find that $r_{23}$ meets the requirements, and the sample vector is extended to $(r_{11}, r_{23}, -)$.

The search now goes to $R_3$. As seen in Fig. 5, $r_{32}$ is the only member meeting the constraints, and so completes the vector $(r_{11}, r_{23}, r_{32})$. The solution set we have found in ${}_3E$ is then

$$S({}_3E) = \{(1,1), (2,3), (3,2)\} \ . \tag{8}$$

At this point in the scan procedure, the position of the scan in ${}_3E$ is saved, and step three is entered.

The third step is to construct the post projection lattice, P, defined by the solution set just found in ${}_3E$ [Eq. (8)]. The index pairs in this set prescribe the posts (parallel to view 3) in the correlation lattice E that are to be used as rows in P. This projection is shown on the left of Fig. 5.

The fourth step--to fix the third index of the solution triplets--is done by a scan of P in the same manner as the scan at ${}_3E$. As shown

in the figure, the solution vector ($r_{13}$, $r_{22}$, $r_{31}$) is the only one possible, and a solution set has been found in the correlation lattice E--

$$S(E) = \{(1,1,2), (2,3,2), (3,2,1)\} . \qquad (9)$$

Elements corresponding to this set are made equal to 1 in the "solution lattice" $E_S$.

After the solution is found in P, the scan of P will continue in an effort to find any more that might be present. If other solutions had been found, they also would have been entered into $E_S$.

When the scan for solution sets in the post projection has been exhausted, the program returns to step two to continue the scan of ${}_3E$ from the point where the last solution set was found. If another solution set is found, steps three and four are again used to determine if the solution in ${}_3E$ is the projection of a solution in the three-dimensional lattice. These procedures continue until the scan of ${}_3E$ is exhausted, at which time the solution lattice, $E_S$, will contain the elements of all solutions found in the scan. There is a second possible solution for our sample event, with the resulting solution lattice shown in Fig. 6.

The solution lattice is used as a replacement for the original correlation lattice in order to eliminate all elements that do not contribute to some solution set. This lattice is now checked for elements common to all solutions in the lattice. These triplets are fully determined and need no further testing. Their elements are set equal to 0, their indices are saved to become a part of the final solution, and the required solution length in E is reduced accordingly. In Fig. 6 we see that the triplet (3,2,1) is of this type and can be eliminated from the lattice. This leaves only four elements to be tested by the second test sequence, thus illustrating the power of what seemed to be a rather weak first test that could eliminate only 5 image pairs out of the 27 tested.

## THE MULTIDIMENSIONAL CORRELATION LATTICE

The properties and procedures described thus far for the correlation lattice can be extended to the case of n stereo views requiring an n-dimensional correlation lattice. If $m_i$ topological characteristics are seen in view i, the lattice will have dimensions of $m_1$ by $m_2$ by ... by $m_n$. Solutions in this lattice will be made of L n-tuplets chosen with the same constraints as those used in the three-view case.

In this general case, the tests of image combinations can compare any number of views from 2 to n, and the concept of the pair and triplet test is extended to the more general k-tuplet test, a binary function of image indices from k views.

It is possible to form various "orders" of projections of an n-dimensional lattice. The first-order projection, ${}_kE$, that we used in the three-dimensional case still eliminates only one view, but it produces an (n-1)-dimensional projection lattice. The elements of the ${}_1E$ projection are now written as

$$ {}_1e_{j\ldots k} = \underset{i=1}{\overset{m_1}{.\mathrm{OR}.}}\ e_{ij\ldots k} \ . \tag{10} $$

The second-order projection, ${}_{jk}E$, eliminates two of the views and is formed by projection from the appropriate first-order projection. In this procedure, the projection operation is commutative, i.e.,

$$ {}_{jk}E = {}_j({}_kE) = {}_k({}_jE) \ . \tag{11} $$

In general, it is then possible to discuss the kth-order projection in which k of the views have been eliminated by successive projections.

An (n-k)-tuplet test will use the kth-order projection lattices to determine which image combinations it is to test and to record the results. However, the masking of the projections back into the lattice may be done in different ways. Storage requirements may prevent the keeping of all intervening projections to allow successively higher order masking back into the n-dimensional lattice. It may be more efficient to record the test results directly into the lattice E; i.e., if the value

of the test function is zero, then all elements in E that use the same indices used by the test function are set to zero.

The scanning procedure for the n-dimensional lattice is simply a continuation of the three-dimensional scan. A solution set in the two-dimensional projection lattice, ${}_{3\ldots n}E$, defines the posts in the three-dimensional lattice, ${}_{4\ldots n}E$, to be used in the construction of the post-projection lattice. A solution set in this lattice then defines posts in the four-dimensional lattice, and a second two-dimensional post projection is constructed. This continues as far as the selection of posts from the n-dimensional lattice E in order to determine the final index of the n-tuplets forming the solution set.

## SUMMARY

The correlation lattice and the iterative backtrack scheme of scanning for solution sets of matching triplets were used as the basis for a code written in the Fortran IV language to match track images of nuclear-particle events. This code has been running on the IBM 7094 for over a year, and timing studies show that it can match images of events with seven images in each view in less than one-half a second, with most of that time devoted to computation for the various image-combination tests.

As developed, the program is divided into two logically separate sections (test and logic) with a simple interface. The tests are represented as Fortran logical functions which have values ".TRUE." or ".FALSE." corresponding to the possibility or impossibility, respectively, of matching the image combination being tested. This division allows the tests to be developed independently of the logic section of the code and permits easy development of new programs using the same solution-finding technique.

The literature offers several examples of the use of backtrack programming.[1,2,3] As Golomb and Baumert so candidly state in their excellent summary of the technique,[1] "Backtrack has been independently 'discovered' and applied by many people." We regret that we are a member of those ranks. However, the notion of using backtrack in an iterative sense as we have done seems to be new. We feel that backtrack in any form may offer other people a very useful tool and hope that wider publication of the method will result in fewer independent discoveries.

ACKNOWLEDGEMENTS

We thank Dr. Frank Solmitz and Dr. Margaret Alston for their suggestions and encouragement during the development of the concepts and methods of the correlation lattice.

This work was performed under the auspices of the U. S. Atomic Energy Commission.

## REFERENCES

1. Solomon W. Golomb and Leonard D. Baumert, "Backtrack Programming," Journal of the Association for Computing Machinery, vol. 12, no. 4, pp. 516-524 (October, 1965).

2. R. J. Walker, "An Enumerative Technique for a Class of Combinatorial Problems," in Proceedings of the Tenth Symposium in Applied Mathematics of the American Mathematical Society, vol. 10, American Mathematical Society, Providence, Rhode Island, 1960, pp. 91-94.

3. J. D. Swift, "Isomorph Rejection in Exhaustive Search Techniques," in Proceedings of the Tenth Symposium in Applied Mathematics of the American Mathematical Society, vol. 10, American Mathematical Society, Providence, Rhode Island, 1960, pp. 195-200.

## FIGURE CAPTIONS

Figure 1. A photograph and sketches of a nuclear-particle interaction.

Figure 2. The initial correlation lattice. (The solid disc ● represents a value one; an open disc ○ represents zero.)

Figure 3. The projection lattices containing the test results ready to be masked into the correlation lattice.

Figure 4. The correlation lattice after masking. Dashed lines show the two remaining solutions.

Figure 5. The lattice and projections used in the scanning procedure. The solution in ${}_3E$ defines posts in E to be used as rows in P.

Figure 6. The solution lattice $E_S$ showing the two remaining solutions.

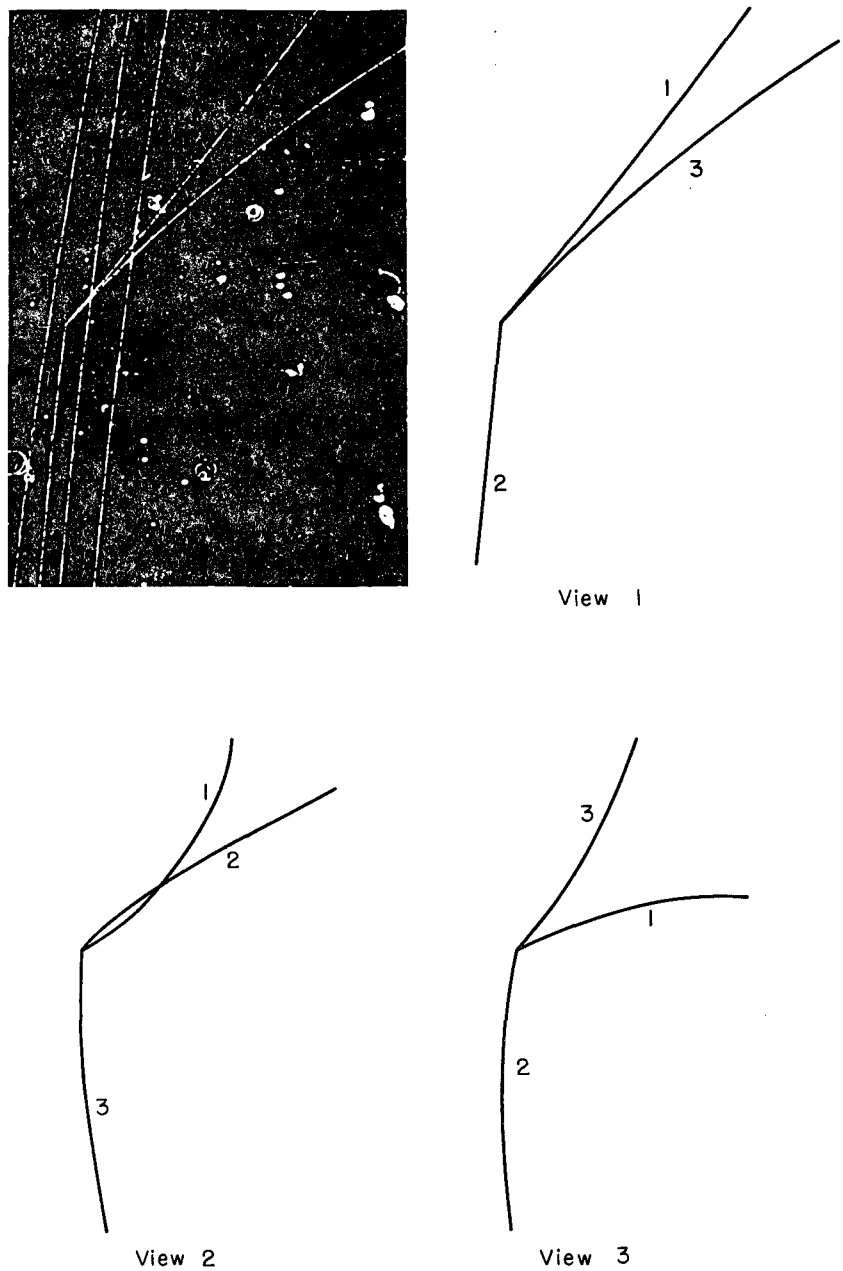
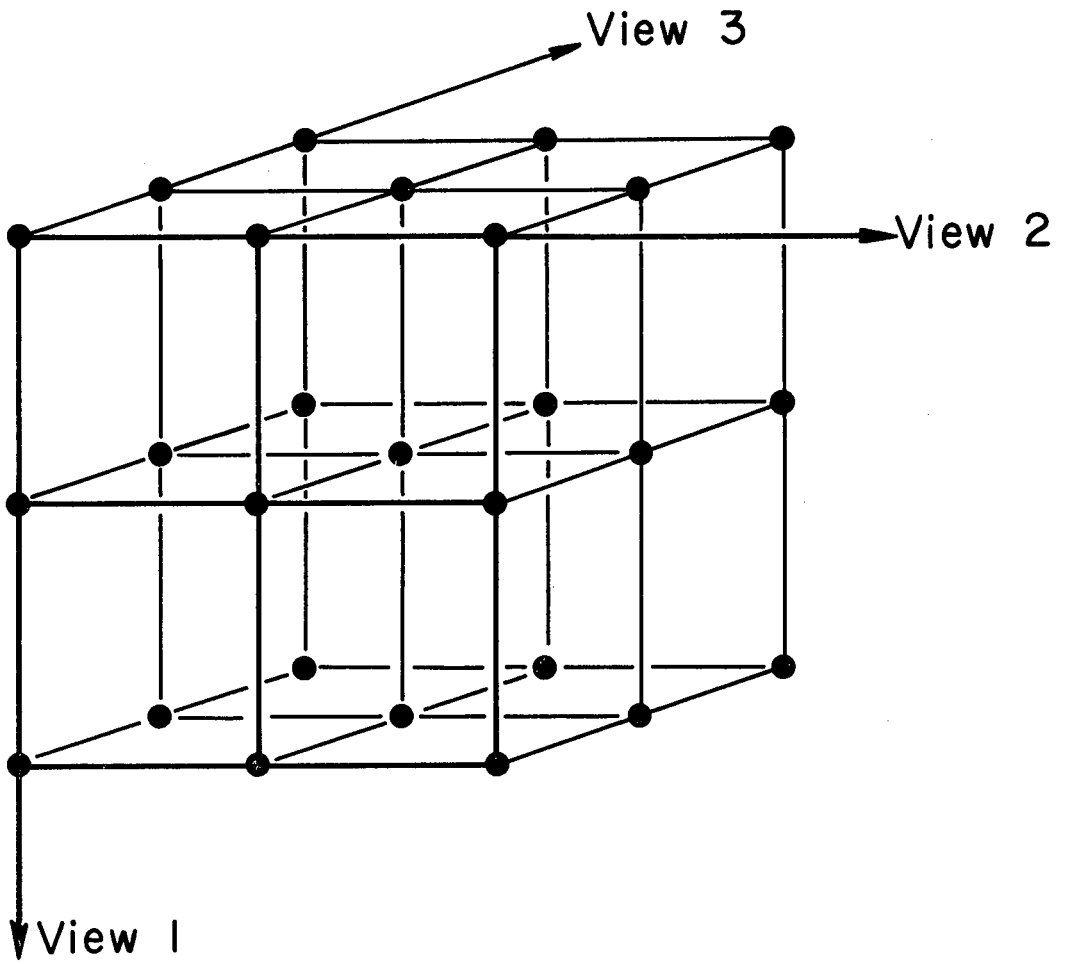
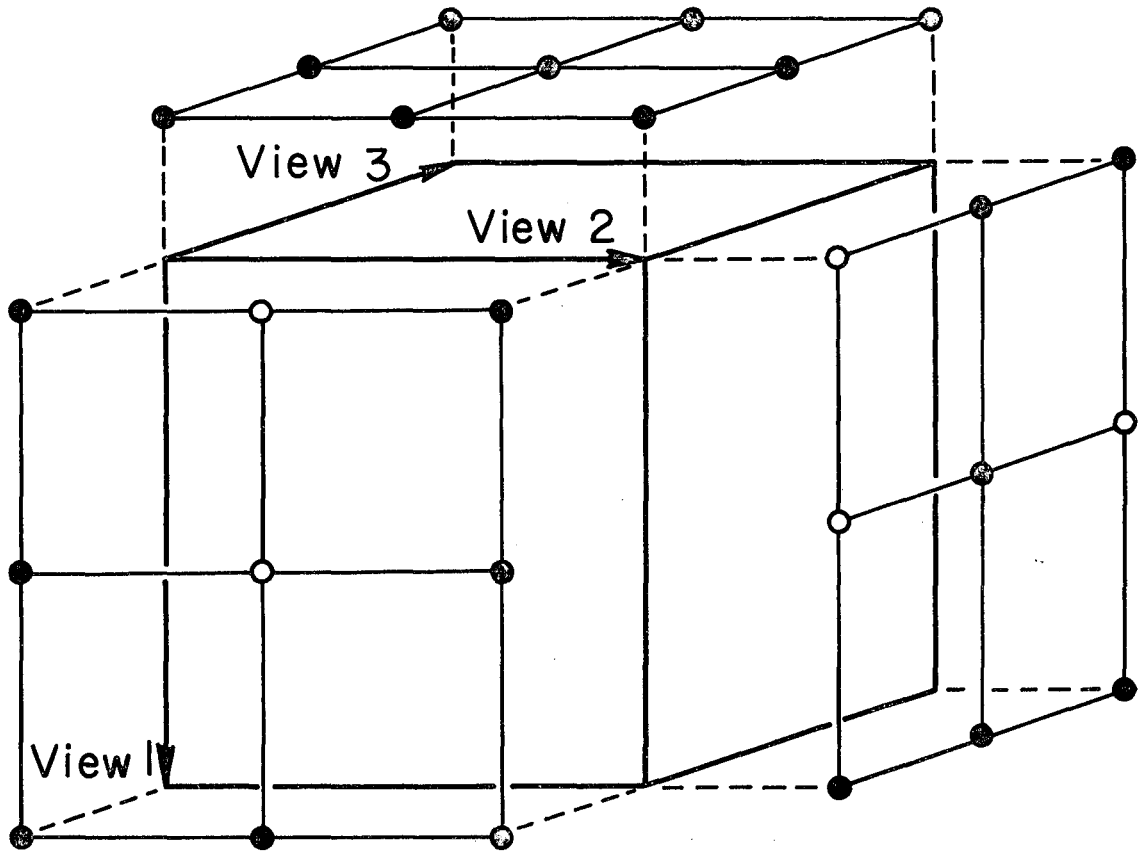



Fig. 1



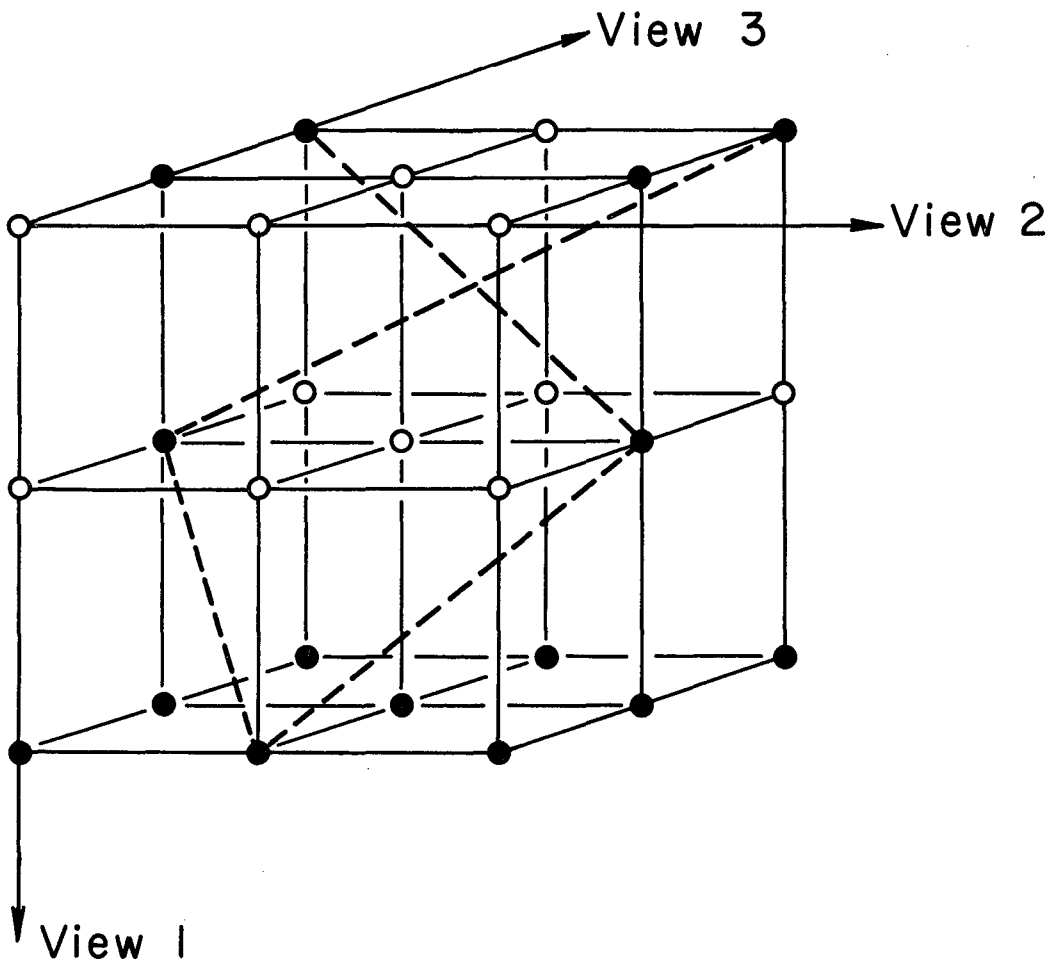

MUB-9404

Fig. 2



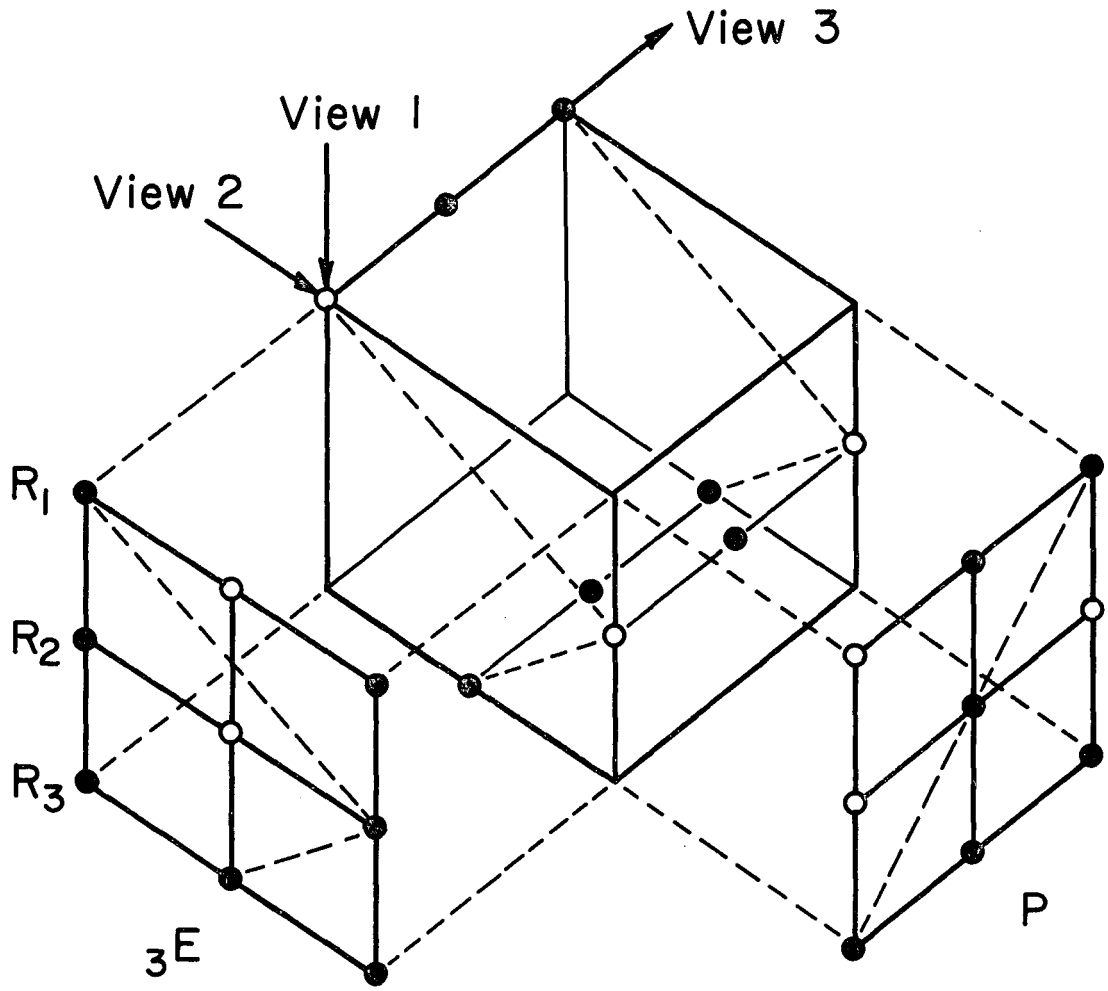

MUB-9403

Fig. 3



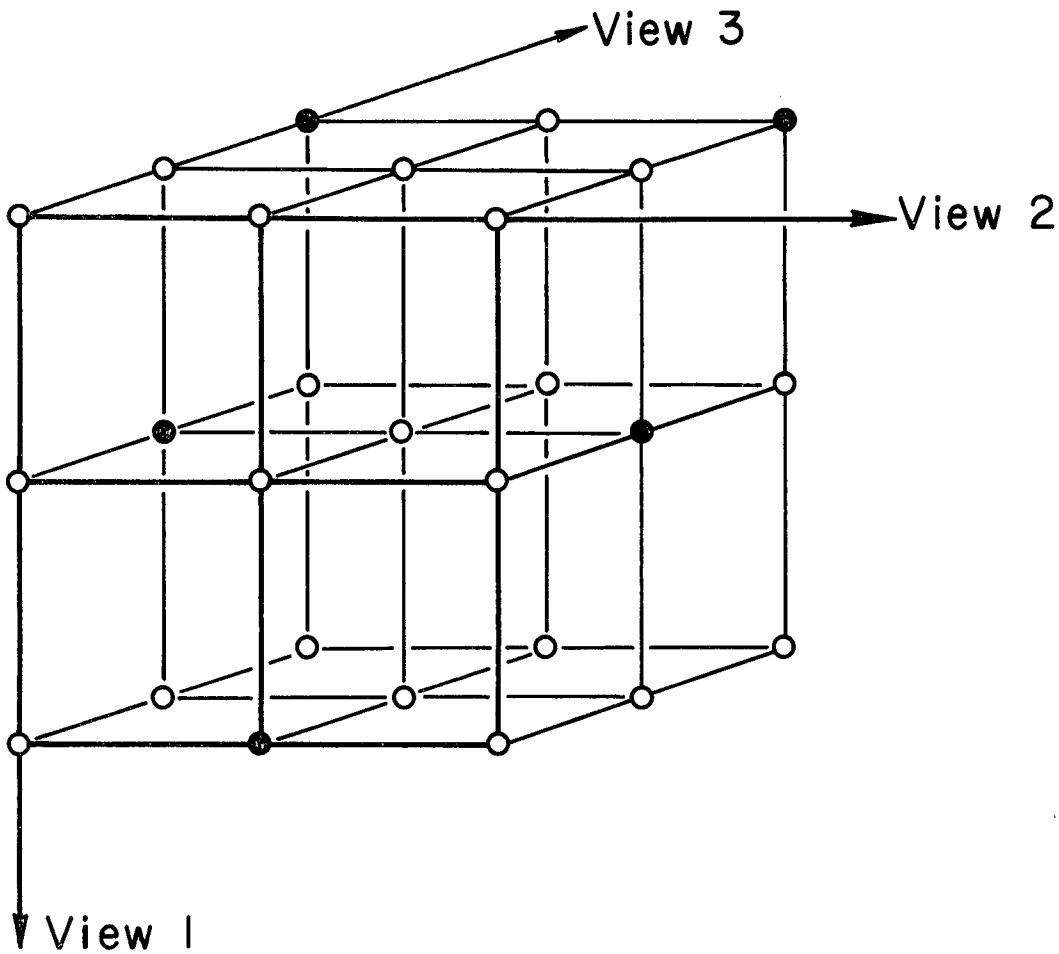

MUB-9402

Fig. 4




MUB-9401

Fig. 5




MUB-9400

Fig. 6

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.