

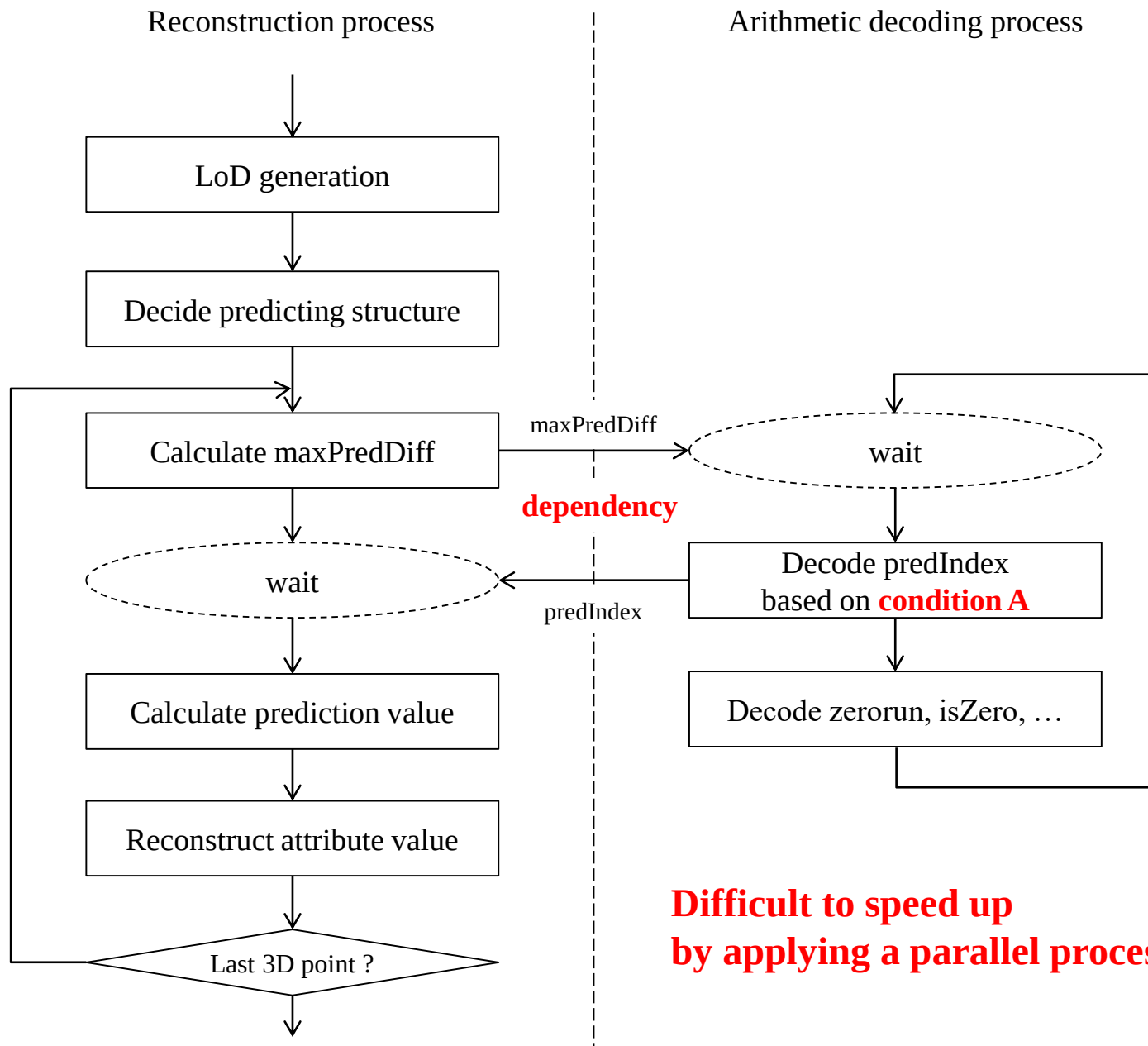
*m49630:*

*[G-PCC] On dependency issue of predicting  
transform in TMC13*

- For Predicting Transform, the multiple predictor candidates could be created and a selected predIndex is encoded into the bitstream
- The predIndex is coded as following,

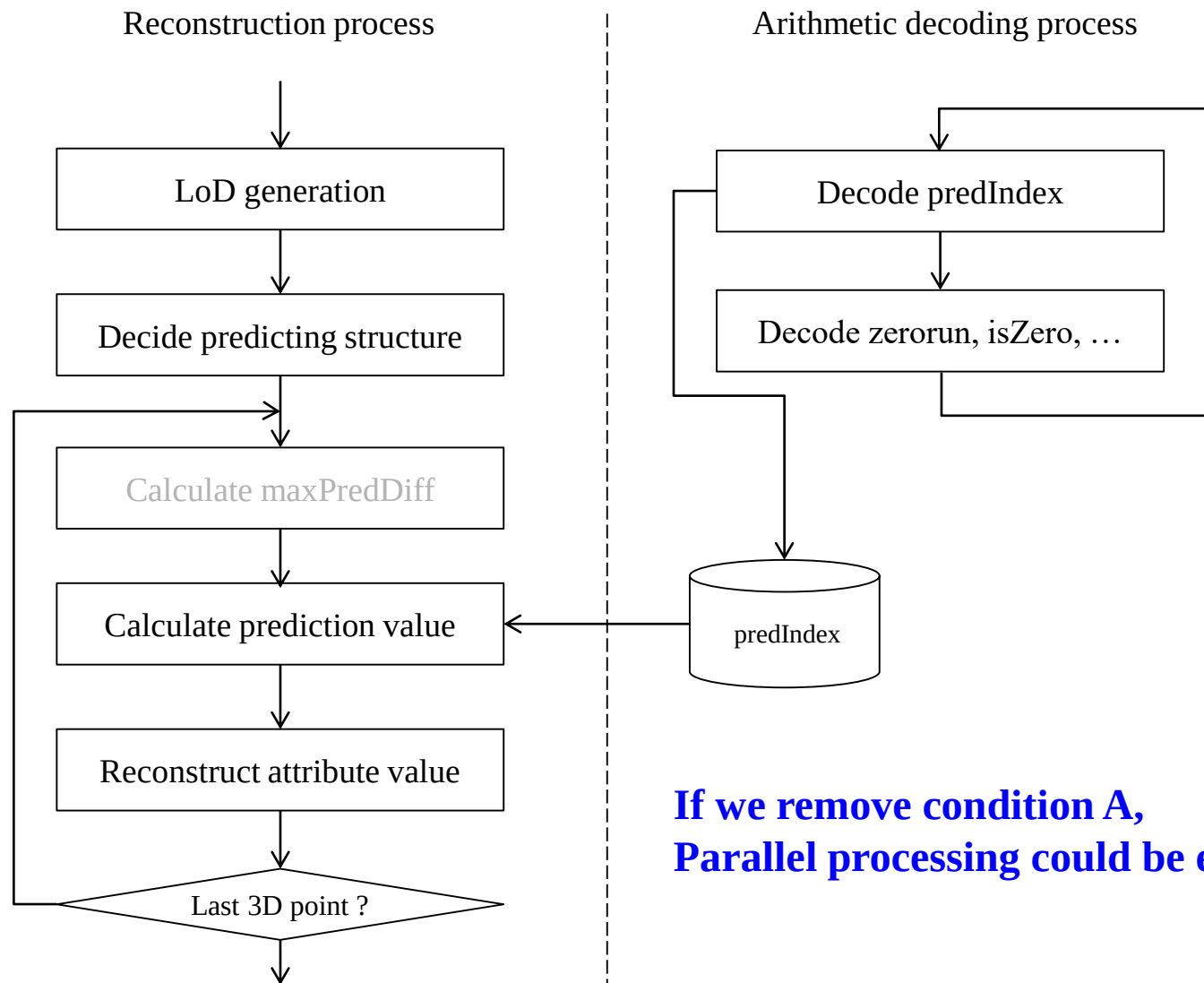
if maxPredDiff > lifting\_adaptive\_prediction\_threshold then, **← condition A**  
    predIndex is coded  
otherwise,  
    predIndex is not coded and it is inferred equal to 0

- maxPredDiff is calculated by using decoded values of neighborhood 3D point  
→ Arithmetic decoding process needs to wait until previous 3D point is decoded in order to decode next predIndex
- We propose one modification to make the specification be much flexible in terms of coding efficiency and parallel processing



# Parallel decoding flow example

4



**If we remove condition A,  
Parallel processing could be easy**

# Issue and Proposal

- To solve this dependency issue, one way is to remove the condition A  
→ It leads significant loss
- Instead of that, one way is to **make the condition A be always true** by setting lifting\_adaptive\_prediction\_threshold properly

if maxPredDiff > lifting\_adaptive\_prediction\_threshold then, ← condition A is forced to be true  
    predIndex is coded  
otherwise,  
    predIndex is not coded and it is inferred equal to 0

- However, there is an issue
  - lifting\_adaptive\_prediction\_threshold is unsigned integer
  - Minimum value of maxPredDiff = 0
  - Impossible to make it in the current specification
- To solve it, we propose to modify the condition A as following,

if maxPredDiff **>=** lifting\_adaptive\_prediction\_threshold then, ← condition A is forced to be true  
    predIndex is coded  
otherwise,  
    predIndex is not coded and it is inferred equal to 0

- During the implementation of this proposal, we found that predIndex is not coded in case of the number of neighborhood 3D point is equal to 1
- We suggest to modify it to make it to be consistent in G-PCC CD text

# Experimental result compared to TMC13v6

- Result of Proposal with bugfix
- Set lifting\_adaptive\_prediction\_threshold to 64 (default value of CTC)

| CW_ai                  | lossless geometry, lossless attributes [all intra]      |           |             |             |
|------------------------|---------------------------------------------------------|-----------|-------------|-------------|
|                        | bpip ratio [%]                                          |           |             |             |
|                        | Geometry                                                | Colour    | Reflectance | Total       |
| Cat1-A average         | 100.0%                                                  | 100.0%    |             | 100.0%      |
| Cat1-B average         | 100.0%                                                  | #VALUE!   |             | 100.0%      |
| Cat3-fused average     | 100.0%                                                  | 100.0%    | 100.0%      | 100.0%      |
| Cat3-frame average     | 100.0%                                                  |           | 100.0%      | 100.0%      |
| <b>Overall average</b> | 100.0%                                                  | #VALUE!   | 100.0%      | 100.0%      |
| Avg. Enc Time [%]      | 91%                                                     |           |             |             |
| Avg. Dec Time [%]      | 92%                                                     |           |             |             |
|                        |                                                         |           |             |             |
| CY_ai                  | lossless geometry, near-lossless attributes [all intra] |           |             |             |
|                        | EtE Hausdorff BD-AttrRate [%]                           |           |             |             |
|                        | Luma                                                    | Chroma Cb | Chroma Cr   | Reflectance |
| Cat1-A average         | -0.1%                                                   | -0.1%     | -0.1%       |             |
| Cat3-fused average     | -0.2%                                                   | -0.2%     | -0.2%       | 0.0%        |
| Cat3-frame average     |                                                         |           |             | 0.0%        |
| <b>Overall average</b> | -0.1%                                                   | -0.1%     | -0.1%       | 0.0%        |
| Avg. Enc Time [%]      | 98%                                                     |           |             |             |
| Avg. Dec Time [%]      | 98%                                                     |           |             |             |

No significant difference

- Dependency issue on predicting transform was reported
- One modification and bug fix was proposed
- The result showed no significant difference after modification
- By applying this modification, it will make the specification be much flexible in terms of coding efficiency and parallel processing
- We suggest including this modification in next CD text and software

# Experimental result compared to TMC13v6

- Result of Proposal with bugfix
- Set lifting\_adaptive\_prediction\_threshold to 0

| CW_ai                  | lossless geometry, lossless attributes [all intra]      |           |             |             |
|------------------------|---------------------------------------------------------|-----------|-------------|-------------|
|                        | bpip ratio [%]                                          |           |             |             |
|                        | Geometry                                                | Colour    | Reflectance | Total       |
| Cat1-A average         | 100.0%                                                  | 94.8%     |             | 96.2%       |
| Cat1-B average         | 100.0%                                                  | #VALUE!   |             | 100.0%      |
| Cat3-fused average     | 100.0%                                                  | 98.6%     | 117.4%      | 101.7%      |
| Cat3-frame average     | 100.0%                                                  |           | 113.8%      | 102.1%      |
| <b>Overall average</b> | 100.0%                                                  | #VALUE!   | 115.2%      | 99.4%       |
| Avg. Enc Time [%]      | 95%                                                     |           |             |             |
| Avg. Dec Time [%]      | 92%                                                     |           |             |             |
|                        |                                                         |           |             |             |
| CY_ai                  | lossless geometry, near-lossless attributes [all intra] |           |             |             |
|                        | EtE Hausdorff BD-AttrRate [%]                           |           |             |             |
|                        | Luma                                                    | Chroma Cb | Chroma Cr   | Reflectance |
| Cat1-A average         | 3.9%                                                    | 3.9%      | 3.9%        |             |
| Cat3-fused average     | 2.8%                                                    | 2.8%      | 2.8%        | 26.0%       |
| Cat3-frame average     |                                                         |           |             | 23.9%       |
| <b>Overall average</b> | 3.8%                                                    | 3.8%      | 3.8%        | 24.5%       |
| Avg. Enc Time [%]      | 105%                                                    |           |             |             |
| Avg. Dec Time [%]      | 99%                                                     |           |             |             |

It shows significant loss for reflectance coding,  
but it can lead some advantage for the decoder like parallel processing

## 7.3.4.3 Attribute slice data syntax

| attribute_slice_data() {                                                                                                 | Descriptor |
|--------------------------------------------------------------------------------------------------------------------------|------------|
| <i>dimension</i> = attribute_dimension[ ash_attr_sps_attr_idx ]                                                          |            |
| <b>zerorun</b>                                                                                                           | ae(v)      |
| for( i = 0; i < pointCount; i++ ) {                                                                                      |            |
| if( attr_coding_type == 0 &&<br>maxPredDiff[ i ] >= lifting_adaptive_prediction_threshold &&<br>MaxNumPredictors > 1 ) { |            |
| <b>predIndex</b> [ i ]                                                                                                   | ae(v)      |
| }                                                                                                                        |            |
| if( zerorun > 0 ) {                                                                                                      |            |
| for( k = 0; k < <i>dimension</i> ; k++ )                                                                                 |            |
| values[ k ][ i ] = 0                                                                                                     |            |
| zerorun -- 1                                                                                                             |            |
| }                                                                                                                        |            |
| else {                                                                                                                   |            |
| attribute_coding( <i>dimension</i> , i )                                                                                 | ae(v)      |
| <b>zerorun</b>                                                                                                           | ae(v)      |
| }                                                                                                                        |            |
| }                                                                                                                        |            |
| byte_alignment()                                                                                                         |            |
| }                                                                                                                        |            |