Multiresolution Modeling

A Very Brief Introduction

Spring 2010

Multiresolution Models: Definition

- A ***multiresolution model*** **consists of a collection of**
  - **Mesh fragments, usually describing small portions of an object with different LOD (*i.e.*, level of details)**
  - **Suitable relations that allow selecting a subset of fragments (according to user-defined quality criteria) and combining them into a mesh that represents the object.**

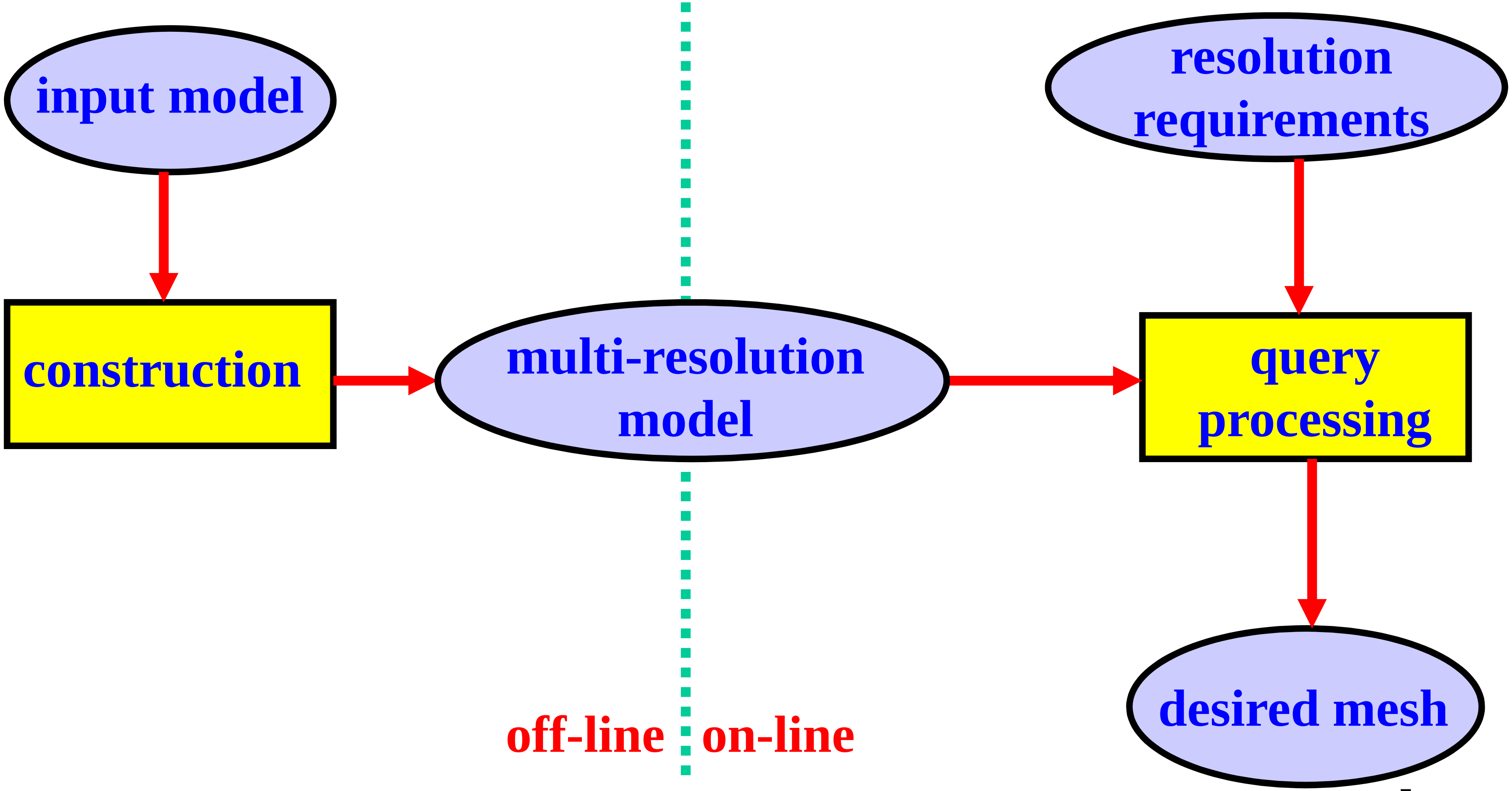
Multiresolution Models: LOD

- **There is no universally accepted definition for LOD, level of details. In general, more faces means more details and perhaps higher accuracy.**
- **Thus, to accurately represent a curvilinear objects, a large number of small faces may be needed (*i.e.*, higher LOD or resolution).**
- **Not all meshes in a scene require very high resolution. For example, back faces of an object or objects very far away from the camera do not need much details.**

Multiresolution Models: Approaches 1/2

- **There are in general two different approaches:**
  - **On-the-Fly (*i.e.*, real time): A new mesh with the desired resolution is constructed from scratch whenever it is needed.**
  - **Off-Line: Design a data structure to collect the mesh fragments and relations in a preprocessing step, and generate the new mesh with a given resolution on-line.**

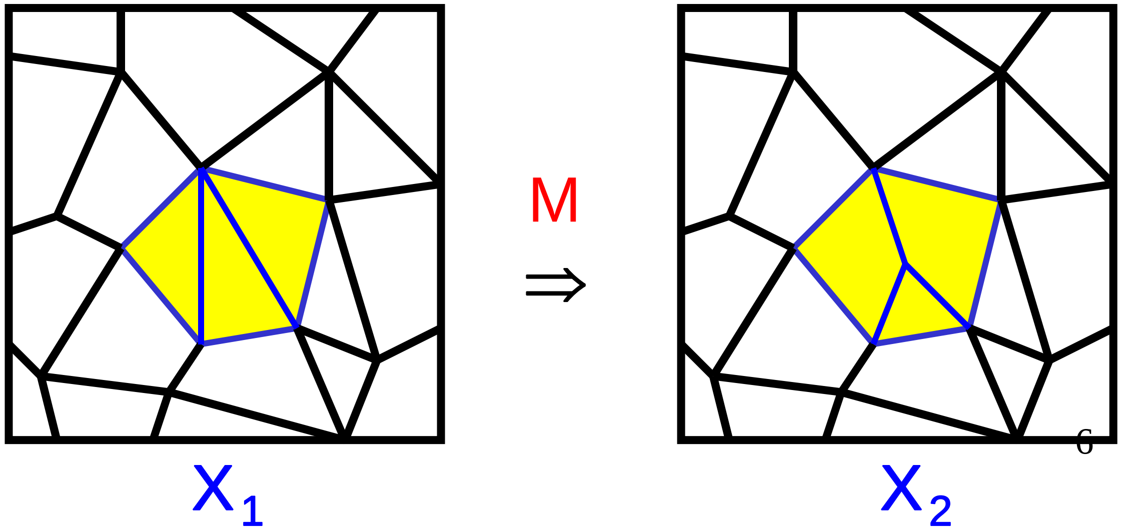
Multiresolution Models: Approaches 2/2



Modifications: 1/3

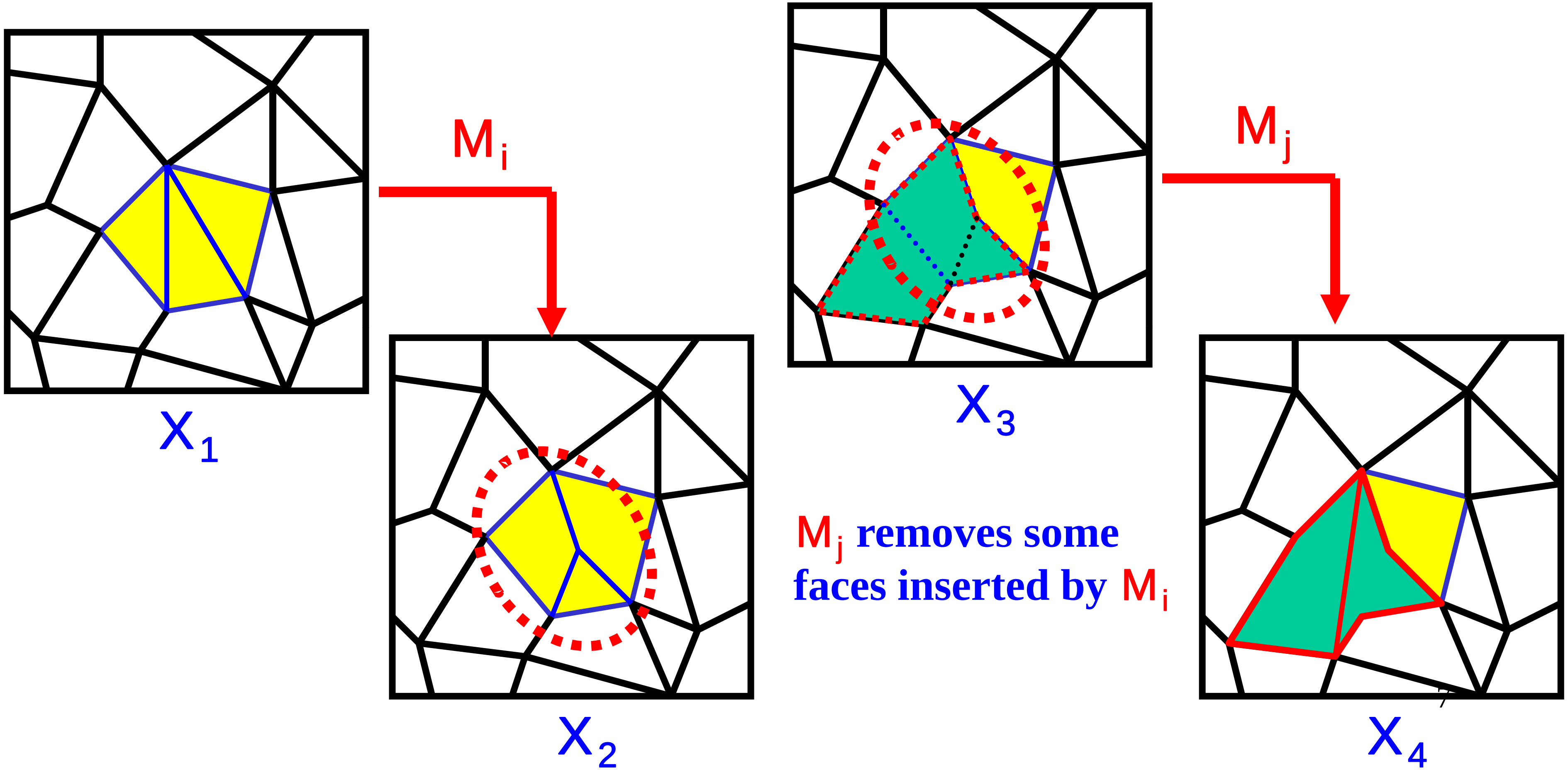
- A **modification** (of a mesh) M is the basic operation of changing a mesh X_1 locally to mesh X_2 written as M: $X_1 \Rightarrow X_2$, or $M = (X_1, X_2)$.
- A modification is a ***refinement*** (***resp***., ***coarsening***) if X_2 has more (***resp***., less) faces than X_1 has.

the yellow region of X_1 is re-tessellated to yield X_2



Modifications: 2/3

- **Not all modifications are independent of each other.**



Modifications: 3/3

- **Given two modifications $M_i=(X_p,X_q)$ and $M_j=(X_s,X_t)$, if modification M_j removes some faces inserted by M_i, we say M_j *directly depends* on M_i, written as $M_i < M_j$.**
- **Given two modifications, they are either independent of each other, $M_i < M_j$ or $M_j < M_i$.**
- **We may apply all possible modifications to a mesh or to its intermediate results until the simplest mesh is obtained.**
- **The modifications may be the Euler operators used in mesh simplifications.**

Multiresolution DAG: 1/3

- **The base mesh X_0, all modifications M_1, M_2, …, M_k, and the dependency relation $<$ together form a multiresolution model $\mathcal{M}=[X_0,\{M_1, M_2, \ldots, M_k\},<]$**
- **Why do we keep track modifications rather than the intermediate results? This is because we can regenerate them from X_0 and the necessary modifications. Storing intermediate results may require very high space consumption.**

Multiresolution DAG: 2/3

- **A directed graph can be constructed as follows:**
  - **The root is X_0, the base mesh**
  - **The directed arcs from X_0 are all modifications applied to X_0**
  - **If there is a modification $M_i < M_j$, draw a directed arc from M_i to M_j.**
  - **In this way, we have a directed acyclic graph, DAG. So, those M_i's are nodes of this DAG!**
- **Why is it a DAG?**

Multiresolution DAG: 3/3

- **With a multiresolution model, we can do the following:**
  1. **Given a “resolution/LOD” requirement, we may perform a depth-first search (DFS) or any search from the root until an intermediate result which satisfies the desired resolution.**
  2. **Selected refinement is also possible. Given a region of a mesh and a “resolution/LOD” requirement, we may perform a search and only “refine” the mesh in the indicated region.**

The End