UCRL-CONF-223462

Simple Common Plane contact detection algorithm for FE/FD methods

O. Vorobiev

August 7, 2006

ICCES
MIami, FL, United States
January 3, 2007 through January 8, 2007

Disclaimer

Simple Common Plane contact detection algorithm for FE/FD methods

Oleg Vorobiev

Lawrence Livermore National Laboratory
L-206, P.O. Box 808, Livermore, CA 94550, USA

Abstract

Common-plane (CP) algorithm is widely used in Discrete Element Method (DEM) to model contact forces between interacting particles or blocks. A new simple contact detection algorithm is proposed to model contacts in FE/FD methods which is similar to the CP algorithm. The CP is defined as a plane separating interacting faces of FE/FD mesh instead of blocks or particles in the original CP method. The method does not require iterations. It is very robust and easy to implement both in 2D and 3D case.

Introduction

Many materials and structures are discontinuous by nature. For example, rock masses include multiple joints and faults, soils are composed of small particles, buildings are composed of bricks etc. Due to big number of discontinuities involved effective contact detection algorithms implemented for distributed-memory parallel computers are very important to model these structures.

Common-plane (CP) algorithm was introduced by Cundall [1] in order to reduce the expensive object-to-object contact detection problem to a less expensive plane-to-object contact problem in DEM. Interacting objects are polygons in 2D case and polyhedral in 3D case. The CP is defined as a plane bisecting the space between the objects. After CP has been determined each object is tested separately for contacts with the common-plane. The algorithm [1] improved in [2] by adding a fast method to identify the right candidates for the plane requires iterations and is designed for DEM.

The algorithm described in the present paper has been designed to model multiple contacts in FE rather than DEM codes. This explicit and, therefore, a more simple Common Plane contact detection algorithm, SCP, has been implemented in a parallel 2D and 3D Lagrange FE code (GEODYN-L) for shared memory machines using MPI. The basic steps of SCP algorithm are described below.

Description of the contact algorithm

To update Common-Planes the following steps are performed:

1) Global space limits are found across all the CPUs. They define global orthogonal sorting bins used on all CPUs.
2) External faces, and existing CPs are sorted into the bins on all CPUs
3) CPUs exchange their bounding boxes, after that the CPUs with overlapping boxes exchange lists of occupied bins. Then the list of the CPUs interacting with the

given CPU is created by selecting CPUs which have common occupied bins. In addition to that the list of common bins is created for each CPU interacting with the given CPU.

4) The list of ghost faces is created that reside on the other CPUs. These ghost faces are copied from the other CPUs using the lists of interacting CPUs and common occupied bins defined in the previous step.
5) New CP entities are looked for with the following requirements: the normals for the separated faces point in opposite directions; faces can not belong to the same element; ghost face pairs are excluded (because they are treated on the other CPU)
6) Once a pair of candidate faces satisfying above mentioned requirements is found, the following steps are performed :

A. coordinates for the face nodes are found at ½ time step using their current coordinates and velocities,
B. trial Point Of Contact (POC) is found as the midpoint of the segment connecting the candidate faces,
C. face projection polygons are found when the nodal coordinates from step 1 are projected onto the CP with the normal defined as an average normal for the faces,
D. the area of contact is found as intersection of the face projection polygons,
E. if the area of contact is positive, one must check if there is an existing CP separating theses faces among the CPs sorted into the current space bin (Note that all CPs store information about the contacts including the IDs of faces they separate),
F. if CP separating the pair does not exist then a new CP is created,
G. the final POC is set to the center of gravity of the contact area polygon,
H. the algorithm described in [1] is used to update CP variables. The number of history variables stored depends on the contact properties. Usually they include the normal and shear displacements, maximum displacements, plastic strain or damage variable and the forces incremented at the plane each time step.

Note that the steps 1-5 are done once per a few cycles depending on the problem. For example, to model wave propagation through the jointed rock mass it can be done just once since the number of contacts stays the same throughout the calculations but for penetration problems it should be done every 2-4 time steps since the contacts change all the time. The steps A-H are done for each cycle to update existing or the new CPs.

Figure 1 illustrates how CP is found in 2D case where the corner of the mesh1 penetrates one of the faces of mesh 2. Two CP are generated acting on face pairs C-A (CP1) and B-A (CP2). The forces calculated at CP are then applied to the nodes of the faces. For normal face-to-face impact of two elements the proposed algorithm gives the same result as the one in [1]. For the case of corner-to-face impact shown in Fig.1 the resultant force is high than the one that would be calculated using traditional CP method in DEM. The sharper is the angle of the corner, the stiffer is the response.

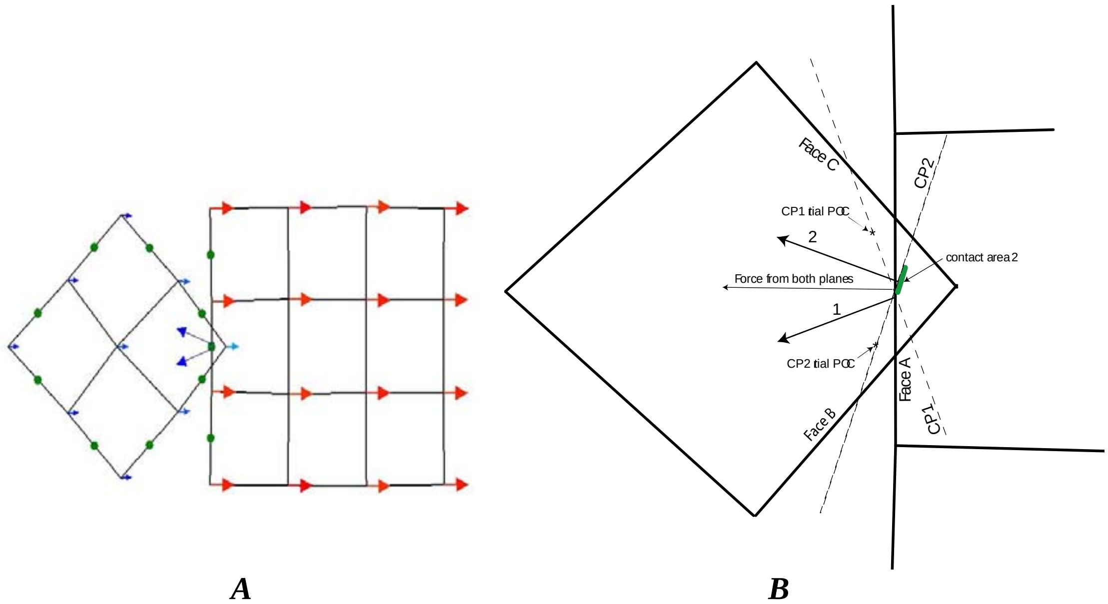

Figure.1*.Common plane determination for 2D impact of two subgrids: A-meshes with velocity vectors, B-the area of contact*

Examples of simulations

2D Projectile penetration through a brick wall

As an illustration of the dynamic contact algorithm in 2D case simulations of a steel projectile penetrating 5 cm thick imbricate wall of concrete at 139 m/s is shown in the figure 2 below. During the penetration of the projectile more and more contacts are activated. Therefore, new Common-planes were looked for every other time step in order to capture new contacts in time.

The concrete response was modeled with pseudo-cap strength model described in [3]. The model includes effects of hardening due to porous compaction and softening due to dilatancy. As material softens its tensile strength approaches zero. The consequence of the effect of softening is the presence of elongated elements containing damaged material shown in Fig.4.

A Coulomb friction law was used for the joints. The load-unload curve included one history-dependent parameter, the maximum joint closure, to describe tangent unloading for the joints.

A bar hitting 3D brick wall

Figure.5 shows results of simulations of a steel 5 m long bar impacting 6.4 x 6.4 x 0.64 m brick wall at 70 m/s. The bar was inclined at 60 degrees. The models mentioned above were used for the concrete and the joints between the bricks.

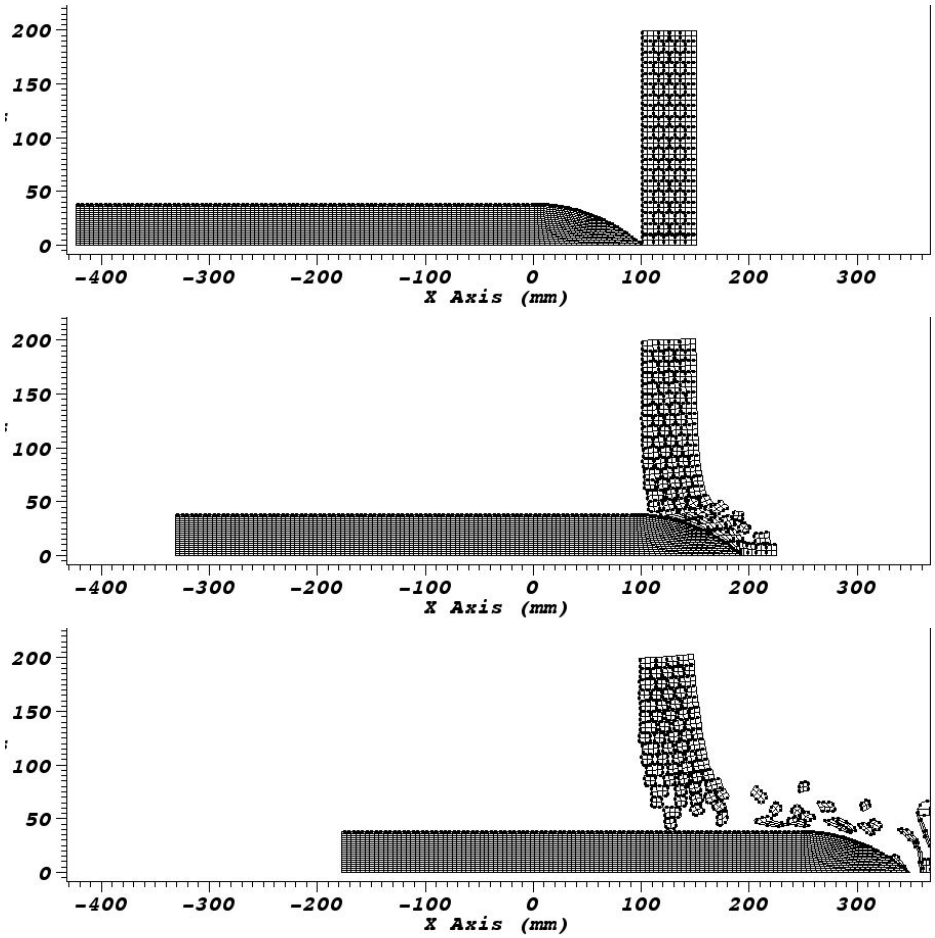

Figure.2 *Numerical mesh and contact face locations (black dots) at various times (t=0 ms is at the top, t=0.8 ms in the middle and t=2 ms at the bottom)*

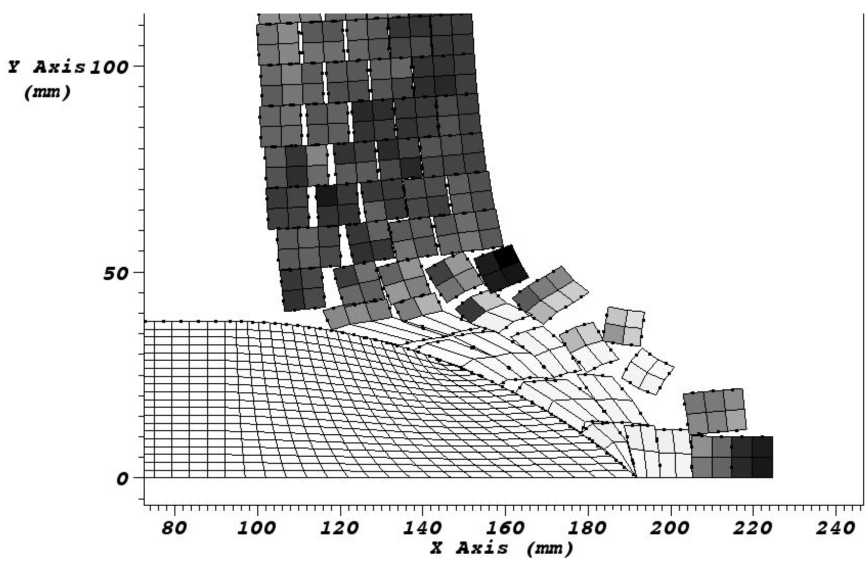

Figure.3 *Numerical mesh and tensile strength contours for concrete blocks at t=0.8 ms. White color corresponds to fully damaged concrete.*

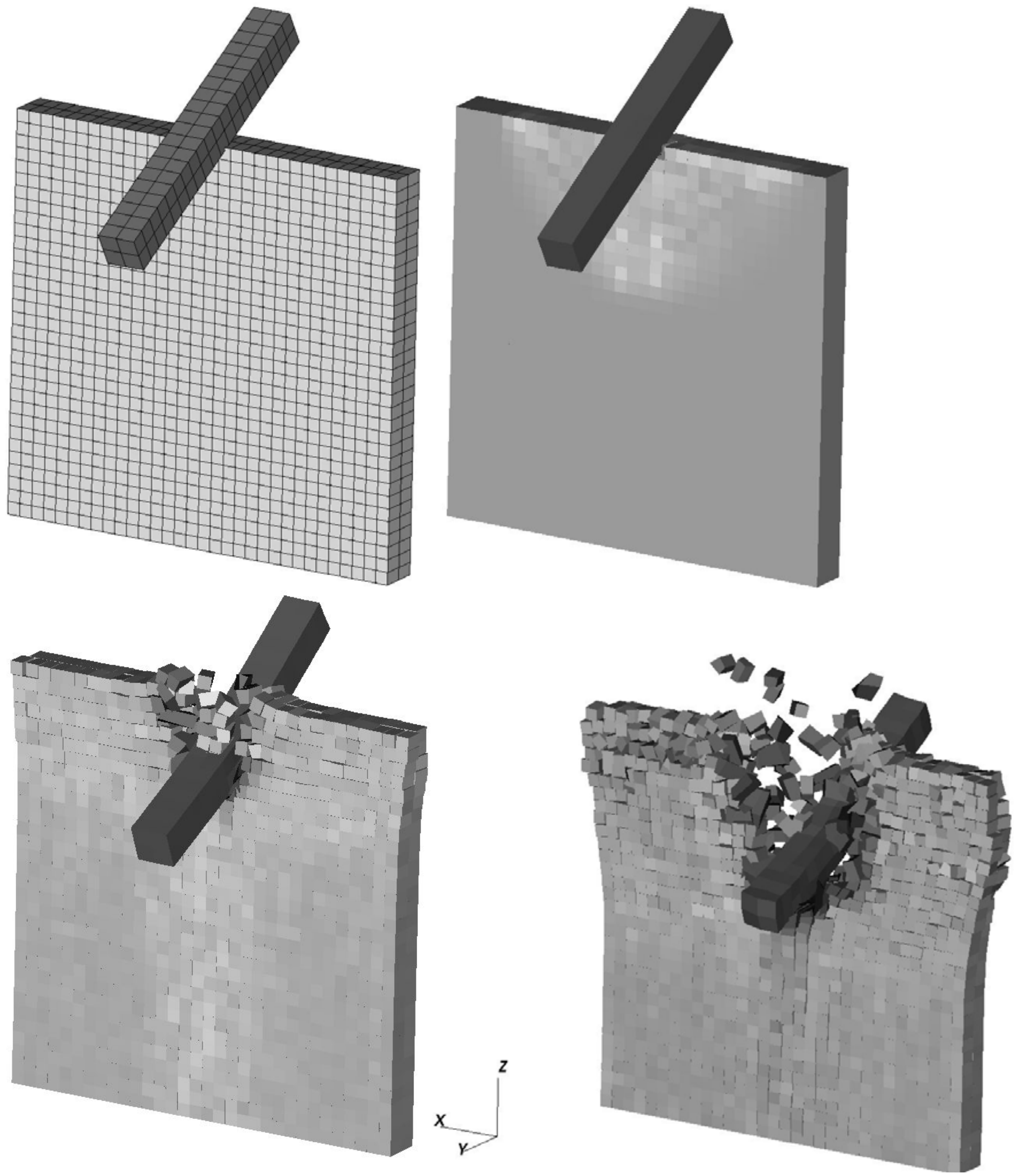

Figure.4 *Numerical mesh at cycle zero and contours of tensile strength for brick wall impacted by a steel bar at times 0.0013 s, 0.0183 s and 0.041 s.*

Conclusion

Simple Common Plane (SCP) contact detection technique is very easy and efficient to model all types of contacts (including vertex-to-vertex, edge-to-vertex, edge-to-edge) Unlike penalty method traditionally used in FE/FD codes, SCP allows advanced friction laws to be applied at contact surfaces with the history-dependent variables.

Parallel SCP algorithm does not provide dynamic load balance for the contacts as, for example, the one described in [4], but it is very generic and easy to implement. The user need not make any decision in advance, for example to specify a master and a slave surfaces.

If both DEM and FE methods are used to solve the problem, then applying SCP as the contact detection technique in FE is very convenient since both CP and SCP methods use common planes separating interacting entities (blocks, particles, faces of the FE mesh etc.). In this perspective SCP can be looked at as an extension of the traditional CP method to include FE.

References

1. Cundall P.A. “Formulation of a three-dimensional distinct element model-part I: a scheme to detect and represent contacts in a system composed of many polyhedral blocks”, *Int.J.Rock.Mech Min.Sci&Geomech Abstr* 25(3) (1998) pp.107-125
2. Nezami E.G. Hashash Y.M.A, Zhao D., Ghaboussi J. “A fast contact detection algorithm for 3-D discrete element method”, *Computers and Geotechnics* 31 (2004) pp.575-587
3. O.Yu. Vorobiev, B.T. Liu, I.N. Lomov and T.H. Antoun “Simulation of penetration into porous geologic media “,*Int.J.Impact.Engn*. in press
4. S.Plimpton,S.Attaway,B.Hedrickson,J.Swegle,C.Vaughan,D.Garner “Parallel transient Dynamics Simulations: Algorithms for Contact Detection and Smoothed Particle Hydrodynamics”, *J.Par.and.Dist.Comp* 50 (1998) pp.104-122

This work was performed under the auspices of the U. S. Department of Energy by University of California, Lawrence Livermore National Laboratory under contract W-7405-Eng-48