碰撞检测

1 Ogm 与 Obstacle 检测方法

表 1 ogm 与 obstacles 检测方法

序号	Ogm 检测	Obstacle 检测
1	Opencv	Opencv
2	点在 polygon 内	SAT(分离轴)
3	轮廓六分圆近似	GJK

2 map 及 path 设置

Map：100m × 100m

Path：由（0.0，0.0）到（100.0，100.0）的一条直线，共 200 个 path point

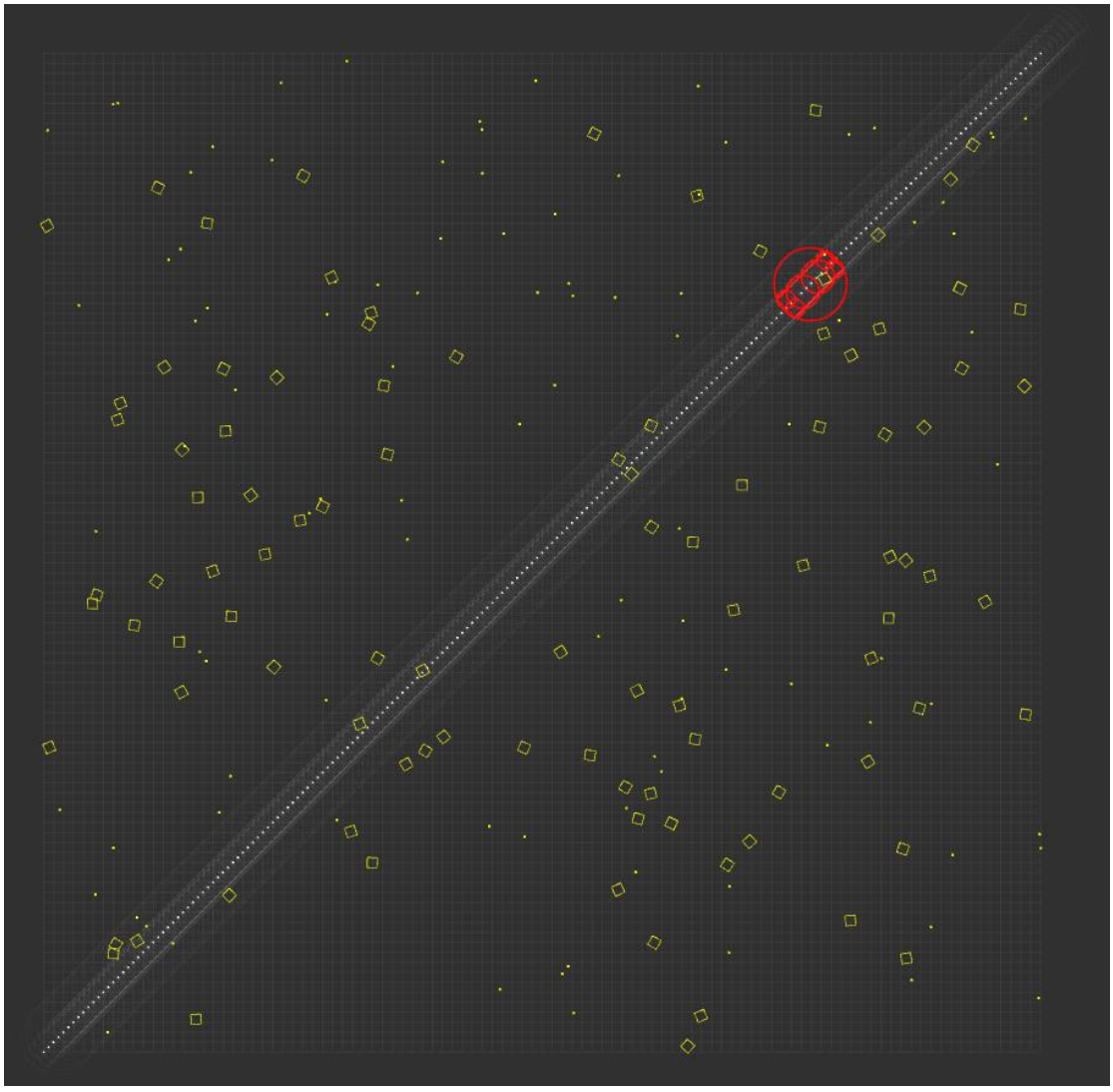

图 1 Collision generate at the 152th of the path with ogm!

Path point：白色圆点

Ogm：黄色圆点

Obstacle: 黄色方框

碰撞点：红色矩形与圆

Path point 上车辆位置：浅白色矩形与圆

3 ogm 检测对比

3.1 碰撞时检测结束

方法：遍历每个 path point，在每个 path point 遍历 ogm，有碰撞即退出

性能：circle >> polygon >> opencv

精度：polygon >> opencv >> circle

表 2 ogm 检测时间对比（碰撞后检测结束）

Ogm	opencv	polygon	circle
10	246.44134	1.6583337	0.2394694
100	46.798352	1.4663385	0.119126
1000	13.994636	0.0612736	0.008881
10000	1.6400916	0.3769269	0.0526028

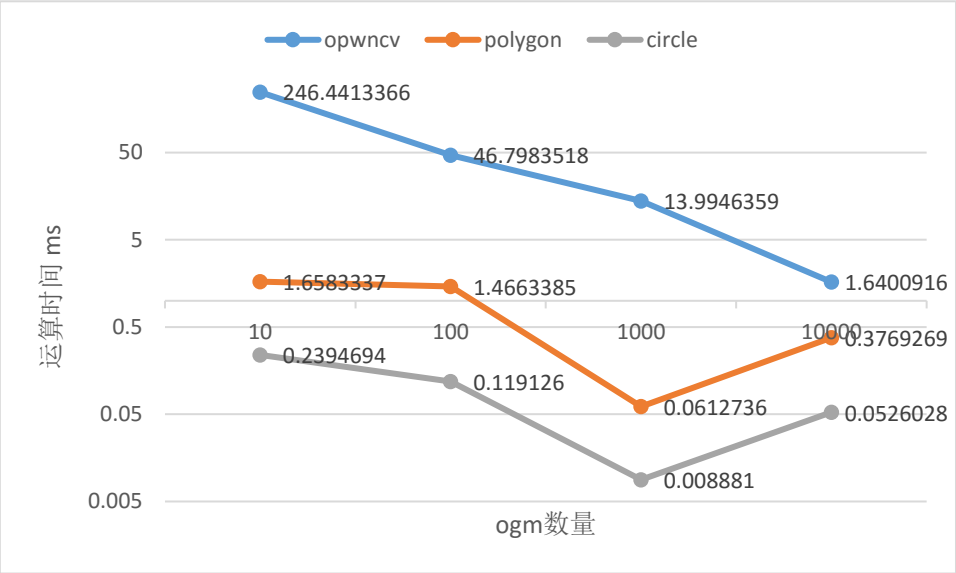

图 2 ogm 检测时间对比（碰撞后检测结束）

由于使用车辆轮廓六分圆近似，其在 x 方向有 8%的误差，在 y 方向有 29%的误差，因此在 ogm 数目为 100 时，在第 18 个 path point 检测到碰撞。

表 3 ogm 检测发生碰撞 path point 位置

ogm	opencv	polygon	circle
10	152	152	152
100	152	152	18
1000	1	1	1
10000	1	1	1

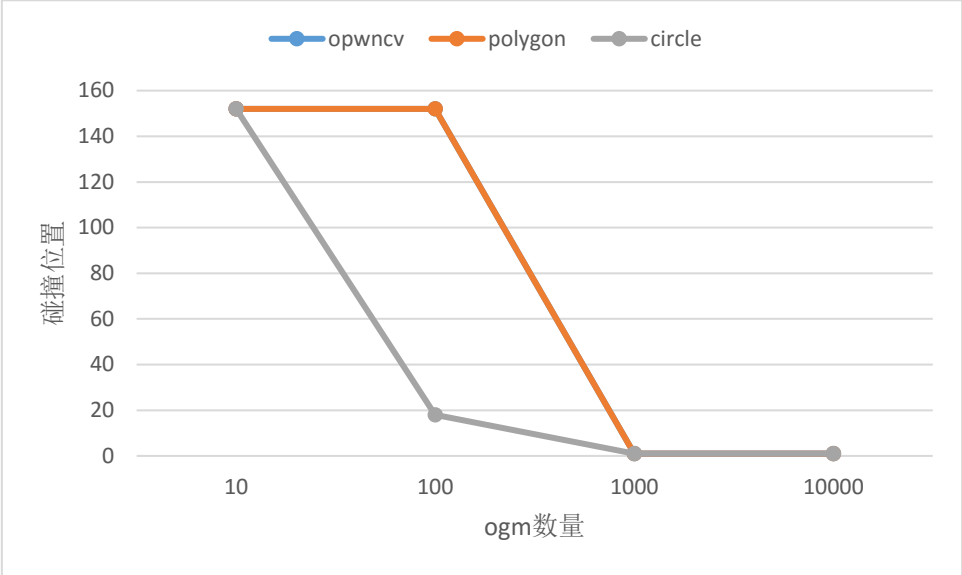

图 3 ogm 检测发生碰撞 path point 位置

3.2 碰撞时检测继续（相当于最恶劣情况）

方法：遍历所有 path point 和 ogm 进行检测；

性能：circle >> polygon >> opencv　n < 5000；

circle >> opencv >> polygon　n > 5000；

表 4 ogm 检测时间对比（碰撞后检测继续）

Ogm	opencv	polygon	circle
10	222.0213	1.238256	0.202343
100	214.9353	4.934635	0.748607
1000	212.0938	43.25367	5.843604
10000	212.0728	394.4772	53.74066

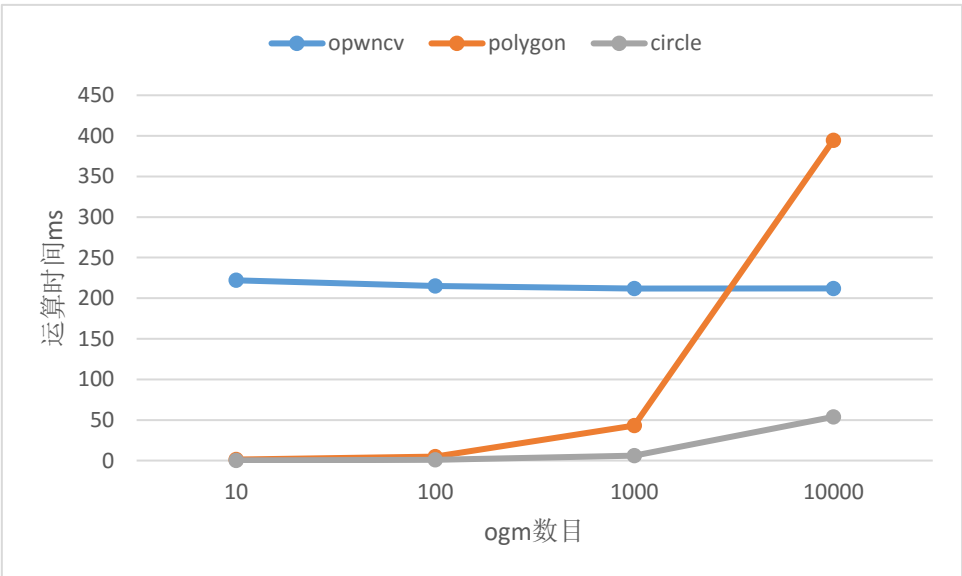

图 4 ogm 检测时间对比（碰撞后检测继续）

此时，opencv 检测性能较为稳定，是因为 opencv 性能主要与 cv::mat 的大小相关。当 ogm 数量为 100 时，地图 100m×100m 不变，改变 opencv 的分辨率，即改变 v::mat 的大小

时，结果如下：

表 5 opencv 检测性能

分辨率	1	0.1	0.01
时间(ms)	0.92148	219.5354	25932.83

4 obstacle 检测

4.1 碰撞时检测结束

方法：遍历每个 path point，在每个 path point 遍历 obstacles，有碰撞即退出

结论：性能：gjk == sat >> opencv

精度：gjk == sat >> opencv

表 6 obstacles 检测时间对比（碰撞后检测结束）

Obstacles	opencv	sat	GJK
10	220.97117	2.9642506	3.1368208
100	31.63754	4.1031606	3.6198686
1000	2.193785	0.5994713	0.5373543
10000	11.669729	3.056633	2.7718017

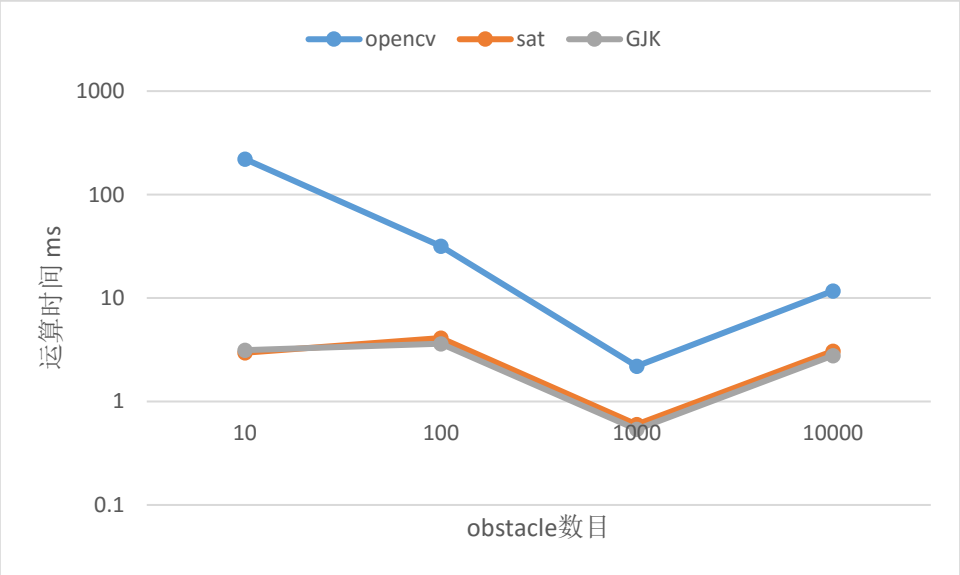

图 5 obstacles 检测时间对比（碰撞后检测结束）

在随机生成的 obstacles 中，三种方法检测到的碰撞位置一致。

表 7 obstacles 检测发生碰撞 path point 位置

Obstacles	10	100	1000	10000
Path point	No Collision	16	1	1

4.2 碰撞时检测继续（相当于最恶劣情况）

方法：遍历所有 path point 和 obstacles 进行检测；

性能：gjk > sat >> opencv n < 8000；

opencv >> sat == gjk n > 8000；(怀疑此处 benchmark 有问题，打印 ros::time 与此不一致)

表 8 obstacles 检测时间对比（碰撞后检测继续）

Obstacles	opencv	sat	GJK
10	217.2471	2.89449	2.881602
100	211.265	28.13448	26.07029
1000	226.5912	265.9454	275.0705
10000	46.6027	2739.34	2694.859

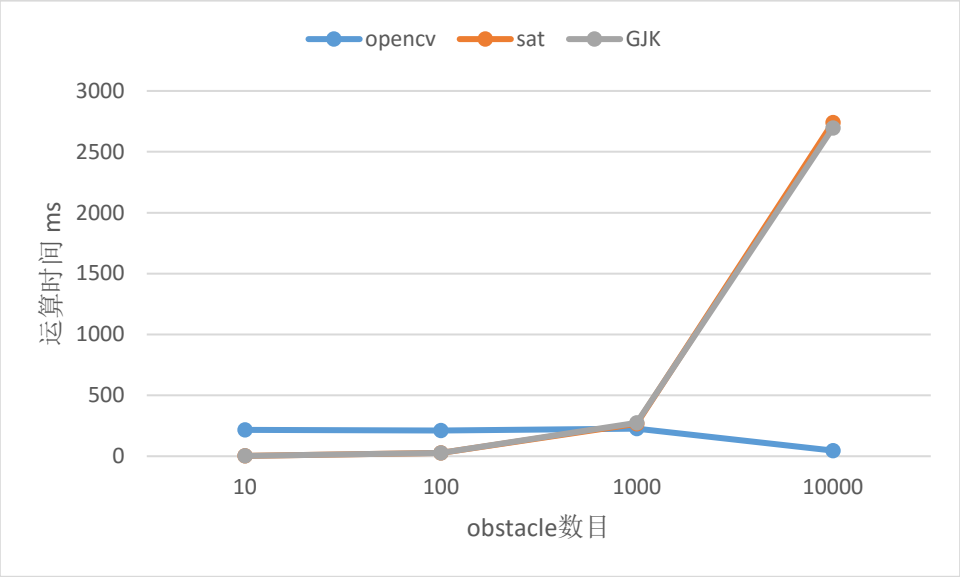

图 6 obstacles 检测时间对比（碰撞后检测继续）

4.3 sat 与 gjk 对比

Obstacles 边数越多，gjk 越有优势。

表 9 SAT 与 GJK 运算时差

Obstacles	10	100	1000	1000
四边形	0.01	2.88	-9.13	275.07
五边形	0.922311	8.972731	87.17821	494.7376
六边形	1.106028	14.63841	133.9084	919.7992

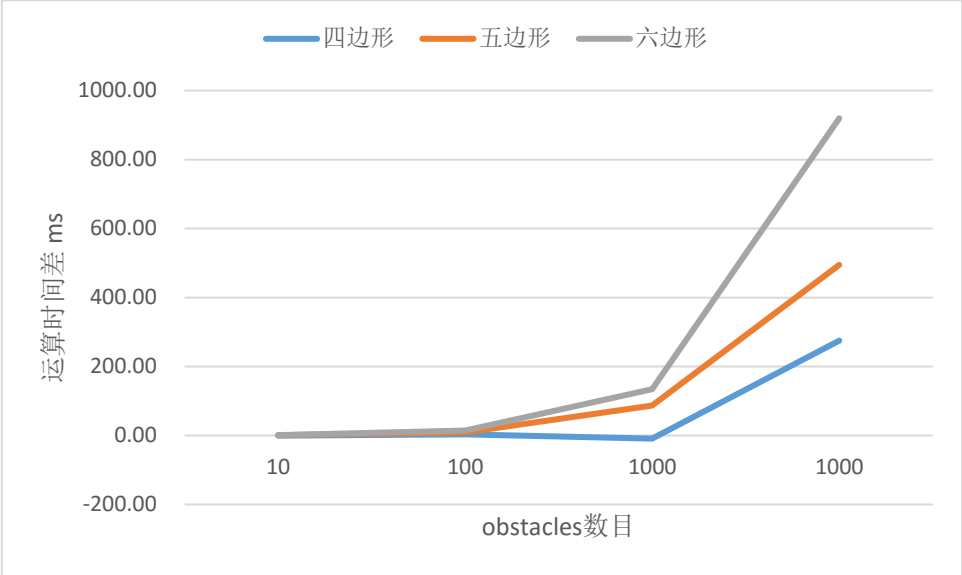

图 7 SAT 与 GJK 运算时差

4.4 sat 简化

由于车和障碍物为矩形，sat 检测轴可以简化为 4 个。

表 10 简化 sat 检测时间对比（碰撞后检测结束）

Obstacles	sat	simple
10	2.964251	0.190093
100	4.103161	0.323203
1000	0.599471	0.011264
10000	3.056633	0.01533

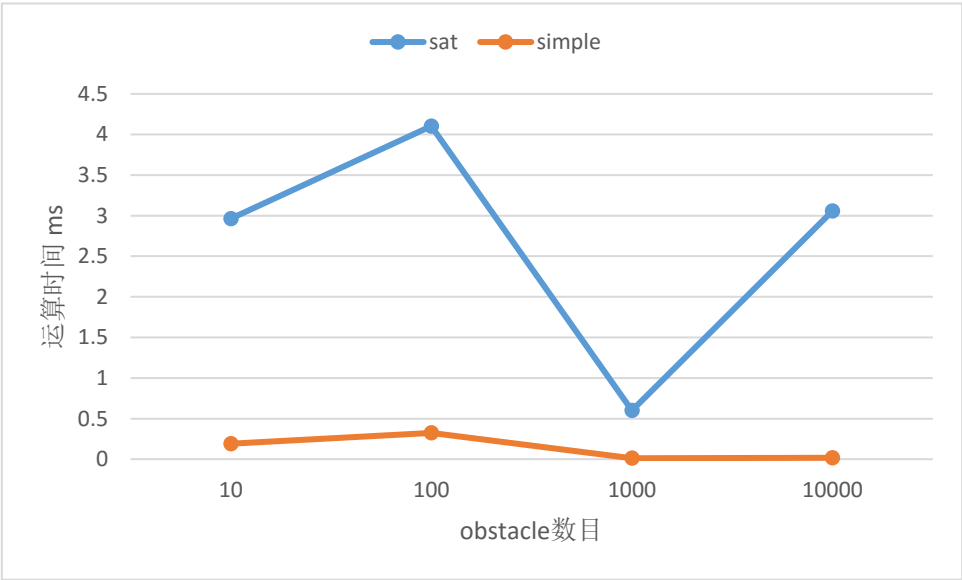

图 8 简化 sat 检测时间对比（碰撞后检测结束）

5 轮廓六分圆近似误差

① 矩形长度对横向误差影响较大

② 矩形宽度对纵向误差影响较大

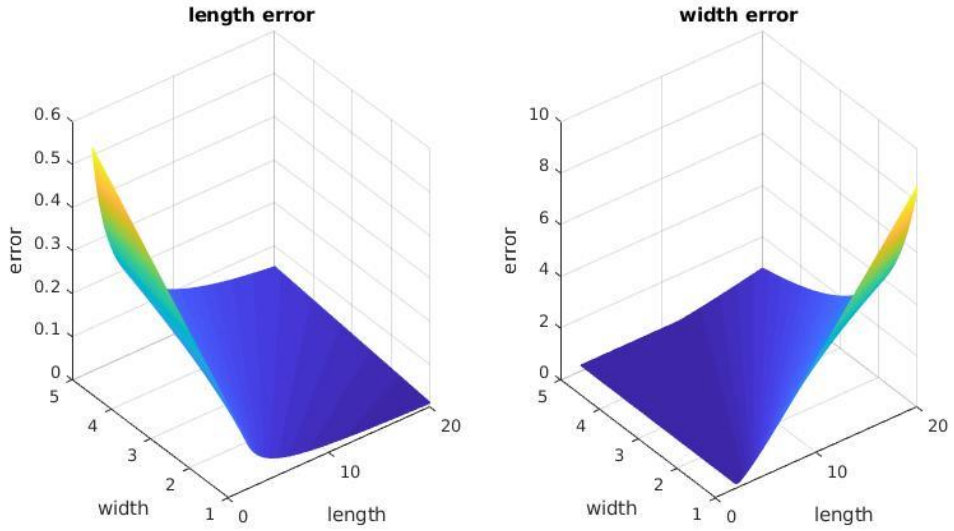

图 9 长和宽对横纵向误差的影响

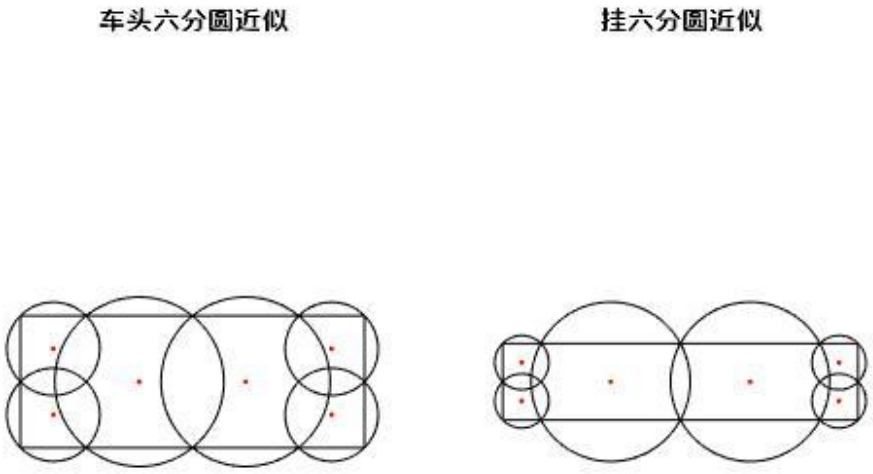

图 10 重汽车头和挂六分圆近似示意

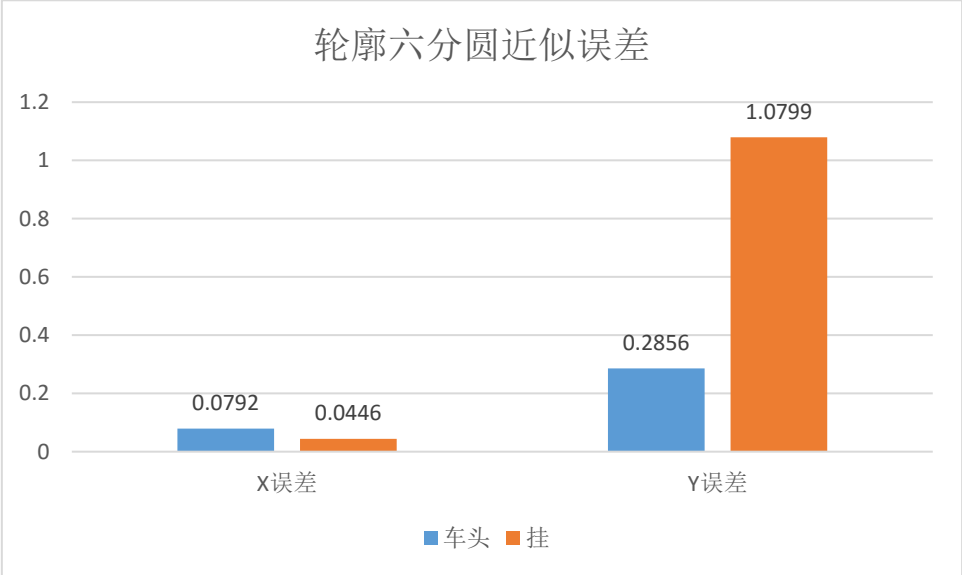

图 11 重汽车头和挂六分圆近似误差

表 11 ogm 检测时间对比（碰撞后检测继续）

Ogm	opwncv	polygon	circle	sat
10	122.4673	0.874554	0.165156	0.112143
100	119.0885	1.07097	0.519425	0.380477
1000	124.535	4.660692	4.099576	3.204693
10000	124.3609	40.00576	40.01218	33.48406

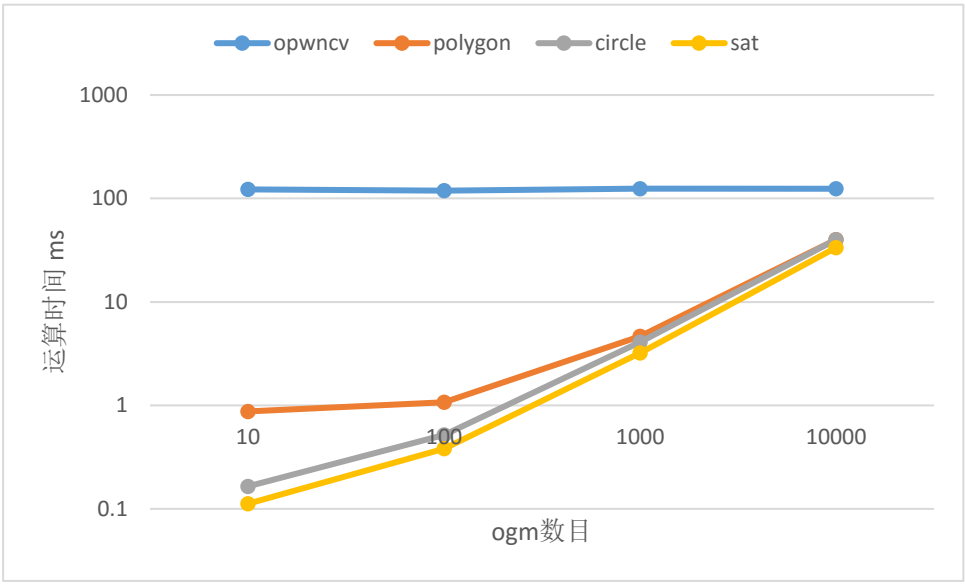

图 12 ogm 检测时间对比（碰撞后检测继续）