nomo17k's the Show feedback from obsessing over MLB11

Part I. Player attribute ratings dynamic range issue

I like recreating players of very unique abilities and let them play in CPU vs. CPU settings to see how they fare, comparing to what they have done in real life. This often means maxing out/minizing down certain player attribute ratings to see how they perform with extreme ratings.

What I found out is the Show could use wider dynamic ranges in converting player ratings into statistics for important attributes. I want to point this out to you the devs, since I've also seen fellow gamers talk with me about how players in the Show, especially pitchers, can look/perform similarly, despite wildly different attribute ratings. For example, they think Mike Pelfrey can often look similar to Roy Halladay, even though their ratings are quite different, when in real life Pelfrey should be mediocre and Halladay dominating. Here, I intend to show some evidence that the Show doesn't necessarily make really great players peform really great and poor players perform really poor.

I've recorded quite a few stats and player ratings from games played using the regular gameplay mode as well as the "sim" mode and compared results to the relevant MLB averages from 2008 – 2011 seasons. to show that the difference among players could be enhanced (therefore more colorful player personalities!!!) by giving wider dynamic ranges to those ratings. Quick summary:

- For gameplay, most player attributes presented here (H/9, HR/9, BB/9, K/9 for pitchers, Contact, PVis, PDisp for batters) should have wider dynamic ranges to produce results; the sim engine is better, but might benefit from fine tunings here and there.
- Batter Power rating has a good dynamic range, so power hitters and weak hitters are represented well already.
- The Show really does a GREAT job of playing MLB games on average! I've looked at many game-generated stats, and can attest that most stats are right there with the MLB numbers, something that I'm really impressed (hence my obsession with the game as a sim gamer).

I don't wish you to take this feedback to mean a wildly huge effort in tuning the game is necessary (certainly please don't break anything, as the game right now already is very well tuned!!). What I think is that **another round of tuning may lead to players having more colorful personalities, where dominant players become really dominant, and sucky players actually do really suck**, etc. I strongly feel that a major reason why very unique players like Mariano Rivera (one-pitch pitcher), Carlos Marmol (extremely wild but really dominating stuff), Ichiro (very solid contact hitter) are NOT reproduced well at all in the game is due to this lack of dynamic ranges I present here.

Also, **please don't take the exact numbers from this document directly to tune the game**... I used my custom sliders mostly (to make the game produce stats in line with MLB averages), so the exact numbers are likely irrelevant when you play with default sliders; it is the dynamic range of ratings to stats mappings that I want to bring to your attention. If you have any questions or something is unclear in this document, please feel free PM me in the OS forum!

Anyways, thank you for lookin at this document from a nutjob, and thank you very much for the game so beautifully crafted!!!

nomo17k

Pitcher H/9 ratings and batting average against

Here are starting pitchers with best/worst H/9 ratings and their batting average against. I cannot fully recover batting average against, since the game does not show things like SF/SH/HBP for pitchers; here I use a proxy as defined by BA = H / (TBF – BB)).)So the real BA against should be slightly larger than what is shown here...)

	rating	sim			gameplay		
Player Name	H/9	H	AB	BA	H	AB	BA
Clayton Kershaw	85	209	811	0.258	57	240	0.237
David Price	82	220	904	0.243	65	237	0.274
Jonathan Sanchez	80	190	799	0.238	62	221	0.281
Jhoulys Chacin	80	184	761	0.242	62	238	0.261
Ubaldo Jimenez	78	214	874	0.245	54	264	0.205
Tim Lincecum	78	179	848	0.211	62	267	0.232
C.J. Wilson	78	212	869	0.244	62	251	0.247
Rich Harden	77	212	796	0.266	76	224	0.339
Edinson Volquez	77	194	842	0.230	62	254	0.244
Jake Peavy	77	216	865	0.250	55	253	0.217
TOTAL:		2030	8369	0.243	617	2449	0.252

	rating	sim			gameplay		
Player Name	H/9	H	AB	BA	H	AB	BA
Scott Olsen	45	222	831	0.267	68	239	0.285
Kyle Lohse	44	254	832	0.305	59	237	0.249
Mark Buehrle	44	249	837	0.297	73	273	0.267
Nate Robertson	44	223	723	0.308	53	233	0.227
Paul Maholm	43	238	802	0.297	72	240	0.300
Trevor Cahill	42	231	865	0.267	70	258	0.271
Kevin Slowey	42	246	850	0.289	68	235	0.289
Doug Mathis	40	243	715	0.340	67	213	0.315
Carlos Silva	36	229	825	0.278	51	244	0.209
Zach Duke	33	223	809	0.276	70	262	0.267
TOTAL:		2358	8089	0.292	651	2434	0.267

The best H/9 group has .252 BA and the worst .267 in gameplay, .243 – .292 in sim. The 2008 - 2011 MLB average has a range from .210 (Michael Pineda) to .300 (Kenny Rogers), so the Show's range is much narrower in gameplay, though sim gets closer.

Pitchers' H/9 ratings definitely should have much more pronounced effects on batting averages in gameplay. For sim, higher H/9 rating probably should lead to slightly less BA against.

Pitcher HR/9 ratings and HR/9 stats

For gameplay, I set CPU Power slider to 4/10, which may affect the number here.

	rating		sim			gameplay	
Player Name	HR/9	HR	IP	HR/9	HR	IP	HR/9
Josh Johnson	80	38	197.0	1.736	7	54.3	1.160
Clayton Kershaw	79	20	205.0	0.878	7	62.0	1.016
C.J. Wilson	78	22	216.0	0.917	6	66.3	0.814
Jaime Garcia	78	24	221.3	0.976	5	53.7	0.839
Tim Lincecum	78	26	224.3	1.043	6	70.7	0.764
Ubaldo Jimenez	78	22	221.0	0.896	6	71.7	0.753
Zack Greinke	77	24	242.0	0.893	5	75.7	0.595
Chris Carpenter	77	24	218.7	0.988	5	67.0	0.672
James McDonald	76	19	190.0	0.900	5	60.3	0.746
Mike Pelfrey	75	31	209.0	1.335	11	59.0	1.678
TOTAL:		250	2144.3	1.049	63	640.7	0.885

	rating		sim			gameplay	
Player Name	HR/9	HR	IP	HR/9	HR	IP	HR/9
Andrew Brackman	48	21	183.3	1.031	5	58.7	0.767
Jordan Zimmermann	48	28	196.0	1.286	4	59.0	0.610
Tommy Hunter	47	29	208.3	1.253	10	61.0	1.475
Aaron Harang	47	30	201.7	1.339	7	61.7	1.022
Rich Harden	47	26	196.7	1.190	5	52.7	0.854
Armando Galarraga	45	27	192.7	1.261	5	48.7	0.925
Ted Lilly	45	17	219.3	0.698	8	62.3	1.155
Kevin Slowey	45	27	207.3	1.172	6	58.0	0.931
Jake Arrieta	42	35	202.3	1.557	11	60.3	1.641
Trevor Cahill	40	23	219.0	0.945	13	65.3	1.791
TOTAL:		263	2026.7	1.168	74	587.7	1.133

The 2008 – 2011 MLB average ranges from HR/9 = 0.50 (Josh Johnson) to 1.59 (Rodrigo Lopez). As for sim, there isn't a much difference between high and low HR/9 rated pitchers at all! With gameplay we see some difference, but not as wide a range as we see in MLB. The Show can obviously use a wider dynamic range for HR/9 ratings!

Pitcher BB/9 ratings and walks

For gameplay, I used 2/10 for CPU Pitcher Consistency, which may affect the number here. (I estimate that this slider setting increases BB/9 by 0.8, which I did by comparing my number here and games played with default sliders.)

	rating	sim			gameplay		
Player Name	BB/9	BB	IP	BB/9	BB	IP	BB/9
Cliff Lee	97	55	221.3	2.236	22	57.0	3.474
Kevin Slowey	96	59	207.3	2.561	22	58.0	3.414
Carl Pavano	94	57	226.0	2.270	14	56.7	2.224
Doug Fister	91	54	194.3	2.501	13	61.0	1.918
Carlos Silva	90	64	202.3	2.847	11	66.3	1.492
Roy Halladay	90	52	245.7	1.905	25	71.0	3.169
Ricky Nolasco	88	58	215.7	2.420	12	56.3	1.917
Joel Pineiro	88	84	196.0	3.857	25	67.7	3.325
Mark Buehrle	87	56	199.7	2.524	17	68.0	2.250
Dan Haren	87	60	212.0	2.547	18	66.7	2.430
TOTAL:		599	2120.3	2.543	179	628.7	2.563

	rating	sim			gameplay		
Player Name	BB/9	BB	IP	BB/9	BB	IP	BB/9
Daisuke Matsuzaka	50	87	194.3	4.029	28	53.0	4.755
Clayton Kershaw	49	82	205.0	3.600	33	62.0	4.790
Scott Kazmir	47	88	208.3	3.802	25	58.0	3.879
Jhoulys Chacin	47	87	194.3	4.029	17	60.3	2.536
Gio Gonzalez	46	92	209.3	3.955	20	64.7	2.784
Andrew Brackman	45	94	183.3	4.615	19	58.7	2.915
Jonathan Sanchez	44	83	209.0	3.574	25	55.0	4.091
Rich Harden	43	86	196.7	3.936	23	52.7	3.930
Edinson Volquez	41	104	216.7	4.320	26	66.7	3.510
Brandon Morrow	40	94	196.3	4.309	25	58.7	3.835
TOTAL:		897	2013.3	4.010	241	589.7	3.678

The 2008 – 2011 MLB average spans a much wider range. For example, 1.29 (Roy Halladay) to 6.02 (Oliver Perez) and 5.92 (Carlos Marmol).

For gameplay: on average the difference among pitchers is there, but the dynamic range is lacking. For pitchers with BB/9 stats, this can probably be corrected somewhat by giving them lower BB/9 ratings . However I don't think we can ever create Hallday/Lee/Maddux even if we use BB/9 = 99. So, maxed out BB/9 rating should lead to much better command, so that BB/9 stats becomes realistic for the best control pitchers.

For sim: the dynamic range is much better, but highest BB/9 rating should still lead to lower BB/9 stats.

Pitcher K/9 ratings and K/9 stats

I did lower Foul Frequency slider to 4/10, which might affect swing through rate. I also lowered Pitcher Consistency slider to 2/10, which might affect stats here.

	rating		sim			gameplay	
Player Name	K/9	K	IP	K/9	K	IP	K/9
Stephen Strasburg	88	221	222.7	8.933	63	66.7	8.505
Tim Lincecum	88	222	224.3	8.906	61	70.7	7.769
Brandon Morrow	85	216	196.3	9.902	48	58.7	7.364
Rich Harden	84	195	196.7	8.924	39	52.7	6.665
Clayton Kershaw	84	211	205.0	9.263	65	62.0	9.435
Yovani Gallardo	83	224	223.0	9.040	57	67.7	7.581
Jonathan Sanchez	83	206	209.0	8.871	35	55.0	5.727
Jon Lester	81	191	217.3	7.910	57	69.0	7.435
Felix Hernandez	80	194	233.3	7.483	52	69.0	6.783
Justin Verlander	80	180	204.7	7.915	65	68.0	8.603
TOTAL:		2060	2132.3	8.695	542	639.3	7.630

	rating		sim			gameplay	
Player Name	K/9	K	IP	K/9	K	IP	K/9
Mark Buehrle	43	90	199.7	4.057	48	68.0	6.353
Carlos Silva	42	103	202.3	4.582	49	66.3	6.648
Doug Mathis	42	83	162.0	4.611	38	51.0	6.706
Jesse Litsch	40	80	195.7	3.680	32	65.7	4.386
Zach Duke	40	87	198.3	3.948	41	67.3	5.480
Brad Bergesen	40	86	196.0	3.949	29	49.7	5.255
Livan Hernandez	40	91	196.3	4.171	45	68.0	5.956
John Lannan	39	83	196.7	3.798	36	62.7	5.170
Jeanmar Gomez	39	74	207.7	3.207	38	60.7	5.637
Aaron Cook	37	89	199.0	4.025	41	68.3	5.400
TOTAL:		866	1953.7	3.989	397	627.7	5.693

The 2008 - 2011 MLB average ranges from 4.25 (Kenny Rogers) to 9.97 (Tim Lincecum).

For gameplay, the Show's range is 5.7 - 7.6 K/9, so it lacks the dynamic range seen in MLB. High K/9 ratings should lead to higher K/9 stats, and low K/9 ratings should lead to lower K/9 stats, perhaps by adjusting swing & miss percentage...

For sim, maybe it is okay as is.

Batter contact ratings vs. batting average stats

Moving on to batters. I use Contact ratings and batting averages against RHPs only.

	rating	sim			gameplay		
Player Name	Contact	H	AB	BA	H	AB	BA
Joe Mauer	99	121	371	0.326	39	120	0.325
Ichiro Suzuki	98	123	466	0.264	46	140	0.329
Josh Hamilton	94	116	367	0.316	33	115	0.287
Miguel Cabrera	93	135	426	0.317	39	141	0.277
Joey Votto	92	114	377	0.302	42	129	0.326
Robinson Cano	92	153	488	0.314	45	131	0.344
Hanley Ramirez	90	122	419	0.291	42	128	0.328
Carl Crawford	89	137	435	0.315	37	147	0.252
Martin Prado	89	127	437	0.291	34	124	0.274
Albert Pujols	89	140	468	0.299	45	143	0.315
TOTAL:		1288	4254	0.303	402	1318	0.305

	rating	sim			gameplay		
Player Name	Contact	H	AB	BA	H	AB	BA
Chris Snyder	38	21	80	0.262	5	27	0.185
Ryan Langerhans	38	5	16	0.312	2	6	0.333
Chris Iannetta	37	26	129	0.202	7	44	0.159
Josh Fields	36	33	163	0.202	12	60	0.200
Adam Everett	35	21	100	0.210	13	43	0.302
Jeff Mathis	35	0	1	0.000	6	16	0.375
Craig Tatum	35	21	94	0.223	4	29	0.138
Jayson Nix	34	84	294	0.286	19	85	0.224
Dave Herman	34	53	204	0.260	20	67	0.299
Kelly Shoppach	32	18	85	0.212	5	34	0.147
TOTAL:		282	1166	0.242	93	411	0.226

The 2008 – 2011 MLB average ranges from .332 (Joe Mauer) to .193 (Jeff Mathis). Looking at a few more players from the MLB averages, it is more like .320 – .220, so the Show covers quite a good range, though the highest Contact rating perhaps could result in a slightly higher batting average. But there aren't as many at-bats sampled here for both sim and gameplay, so some of this could be attributed to small sample sizes. But given both sim and gameplay engines ceils around .300, which might be too conservative.

Batter power ratings and HR/H stats

Power ratings and HR/H stats, both against RHPs only. Using HR/H here because I'm guessing the Power rating dictates how fast the batted ball travels, so if H is a good proxy for line drive clean hits, HR/H should roughly indicate how fast/far the batter can drive the ball.

	rating	sim			gameplay		
Player Name	Power	H	HR	HR/H	H	HR	HR/H
Ryan Howard	99	126	36	0.286	41	14	0.341
Adam Dunn	97	111	31	0.279	45	12	0.267
Prince Fielder	96	146	28	0.192	54	13	0.241
Carlos Pena	96	89	32	0.360	28	11	0.393
Jose Bautista	94	120	23	0.192	36	15	0.417
Russell Branyan	93	101	20	0.198	25	6	0.240
David Ortiz	91	140	44	0.314	49	16	0.327
Josh Hamilton	91	116	31	0.267	33	8	0.242
Jason Heyward	90	104	27	0.260	35	10	0.286
Adrian Gonzalez	90	134	27	0.201	44	11	0.250
TOTAL:		1187	299	0.252	390	116	0.297

	rating	sim			gameplay		
Player Name	Power	H	HR	HR/H	H	HR	HR/H
Nick Punto	28	62	2	0.032	16	1	0.062
Juan Pierre	27	115	1	0.009	55	1	0.018
Nyjer Morgan	27	116	2	0.017	43	0	0.000
Luis Castillo	25	65	3	0.046	11	0	0.000
Angel Sanchez	25	40	1	0.025	11	1	0.091
Cesar Izturis	25	28	1	0.036	11	1	0.091
Andres Blanco	25	36	2	0.056	7	0	0.000
Eric Young Jr	25	41	2	0.049	7	0	0.000
Robb Quinlan	25	45	2	0.044	13	0	0.000
Ruben Tejada	22	10	1	0.100	4	0	0.000
TOTAL:		558	17	0.030	178	4	0.022

The 2008 – 2011 MLB average ranges from 0.228 (Albert Pujols, Ryan Howard, Prince Fielder, Mark Teixeira, Mark Reynolds, all combined) to 0.008 (Chris Getz, Willy Tavaras, Jamey Carroll, Omar Vizquel, Juan Pierre).

The Show has an overall bias toward more power (this despite I'm using CPU power slider set to 4/10 for most of my gameplay data recording).

Both for gameplay and sim, very low power hitters produce a few times more HRs than the MLB average indicates.

It confirms that how we felt there are a lot of HRs generated at default settings. Unless this was intentional, a slight reduction in power production may be desired.

Batter plate vision ratings and K% stats

Using K/PA (strikeouts / plate appearance) stats as a measure of Plate Vision.

	rating	sim			gameplay		
Player Name	PVis	SO	PA	K%	SO	PA	K%
Juan Pierre	99	73	663	0.110	20	226	0.088
Placido Polanco	97	59	677	0.087	26	213	0.122
Jeff Keppinger	97	45	360	0.125	9	94	0.096
Alberto Callaspo	94	57	410	0.139	12	134	0.090
Dustin Pedroia	94	62	641	0.097	30	202	0.149
Alex Cora	90	29	265	0.109	8	83	0.096
Miguel Tejada	89	94	658	0.143	20	213	0.094
Augie Ojeda	89	40	246	0.163	12	72	0.167
Yuniesky Betancourt	88	67	582	0.115	23	188	0.122
Jason Kendall	88	26	271	0.096	10	76	0.132
TOTAL:		552	4773	0.116	170	1501	0.113

	rating	sim			gameplay		
Player Name	PVis	SO	PA	K%	SO	PA	K%
Carlos Pena	24	152	612	0.248	52	189	0.275
Eric Patterson	23	22	99	0.222	10	33	0.303
Ryan Howard	23	163	682	0.239	44	212	0.208
Adam Dunn	21	113	486	0.233	39	170	0.229
Miguel Olivo	20	126	545	0.231	35	167	0.210
Josh Fields	18	68	314	0.217	31	104	0.298
Russell Branyan	11	160	588	0.272	51	182	0.280
Jack Cust	9	123	531	0.232	46	166	0.277
Kelly Shoppach	9	64	232	0.276	18	80	0.225
Mark Reynolds	3	163	646	0.252	54	201	0.269
TOTAL:		1154	4735	0.244	380	1504	0.253

The 2008 – 2011 MLB average for K% ranges from 0.061 (Juan Pierre) to 0.335 (Mark Reynolds).

For both sim and gameplay, high/low PVis rating should results in lower/higher K%.

Batter plate discipline ratings and BB% stats

Using BB/PA (walks / plate appearances) stats as a measure of plate vision rating.

	rating	sim			gameplay		
Player Name	PDisp	BB	PA	BB%	BB	PA	BB%
Jack Cust	99	59	531	0.111	15	166	0.090
Albert Pujols	99	98	731	0.134	34	226	0.150
Kosuke Fukudome	99	25	197	0.127	1	61	0.016
Jason Giambi	99	2	26	0.077	0	2	0.000
Todd Helton	99	65	588	0.111	14	175	0.080
Daric Barton	99	80	628	0.127	19	200	0.095
Chipper Jones	99	77	617	0.125	22	192	0.115
Nick Johnson	99	110	713	0.154	14	227	0.062
Adam Dunn	99	55	486	0.113	22	170	0.129
Ryan Langerhans	99	2	36	0.056	1	12	0.083
TOTAL:		573	4553	0.126	142	1431	0.099

	rating	sim			gameplay		
Player Name	PDisp	BB	PA	BB%	BB	PA	BB%
Howie Kendrick	37	26	587	0.044	16	204	0.078
Miguel Tejada	37	12	658	0.018	18	213	0.085
Chris Stewart	37	7	137	0.051	1	30	0.033
A.J. Pierzynski	35	31	641	0.048	15	212	0.071
Ivan Rodriguez	34	21	503	0.042	16	167	0.096
Yuniesky Betancourt	34	27	582	0.046	14	188	0.074
Jose Lopez	34	25	579	0.043	9	188	0.048
Humberto Quintero	32	11	153	0.072	2	48	0.042
John McDonald	30	11	232	0.047	11	78	0.141
Dave Herman	29	11	299	0.037	6	88	0.068
TOTAL:		182	4371	0.042	108	1416	0.076

The 2008 – 2011 MLB average for BB% ranges from 0.034 (Yuniesky Betancourt) to 0.166 (Jack Cust).

Both for sim and gameplay, the dynamic range is lacking. The reason why SCEA default roster assigns PDisp = 99 for many players is perhaps because of this lack of dynamic range, since in real life many players walk more than PDisp = 99 players in the Show.

For gameplay, the PDisp should have a lot more stronger effect on players' ability to walk (and not walk as well).

For sim, it's much better but even there PDisp = 99 players aren't quite walking as much as they would in MLB.

Part II. My feedback posted in the OS forum

I just copied and pasted them here again, since unfortunately I kept updating my old post and I don't know if CD guys like Pared and nemesis went back to older posts to update their memos before the threads got closed.

There are a lot of good suggestions by other memebers and a lot of mine overlap with them, so please don't bother with this part if it looks too neurotic... Since nobody really mentioned, if anything I'd like Part I above to be my main feedback for the devs this time.

Thank you again!!

Analog control suggestions, improvements or tweaks for MLB 12

Analog Mode: Hitting

Biggest obstacle when using this particular mode:

(1) The analog response for swinging is definitely slower than button (timing/zone). It's expected given you have to move the stick more than you press the button, but this makes it much harder to turn to high/inside pitches. (On the other hand, check swinging is great... so I'd wish to see quicker response without compromising the check swing sensitivity.)

(2) It's so easy to inadvertently "complete" the check swing on slower pitches like curve ball. What I'm talking about is when I intend to swing on a curveball, the hitter doesn't swing (this has been discussed a couple times earlier in the year) but just twitches and does check swing. What causes it is... well let me explain: when the stride is made by pulling R3 down, on a faster pitch, the next motion would be pushing R3 up quickly so the hitter swings normally. On a curve ball (or some other slower pitch), however, in order to stay back R3 stick needs to stay down a bit longer. What often happens is that when holding R3 by thumb, I tend to make slight upward movement which ends up "completing" a check swing. So when I actually push up R3, the interface thinks the swing has already completed and doesn't actually commit to the curveball. I'm still trying to adjust to this but I still have this problem quite often.

Any improvements to the mechanic that could help overcome the above obstacle: Not knowing how things are exactly coded, I don't know how best things can be achieved... but hopefully I explained the issues well enough above to help improve things.

Anything you feel is missing or needs to be taken away from the mechanic to make it function better: Some people have mentioned about the zone-type hitting with analog. That option would be nice. But just like button, we should have options to choose. Timing (R3 just determines swing timing), Zone (L3 for moving PCI), and Default (as is, R3 adjust lateral PCI location). I'm not good at all and still haven't mastered it, but I actually like the idea of having a single stick controlling the whole swing sequence. A lot of hitting is reactionary, so I feel simpler the better (like the current system.... though having control on vertical PCI movement is definitely desired... maybe preloading?? dunno how best can be done).

If applicable, is the visual feedback/graphics ample when using this mechanic. If not what is lacking or needs to be changed? It's quite good, but seeing how many people make fuss over PCI movement not realistic, an option to turn off visual feedback might be nice (partly joking, partly serious...). Especially with online play, I can actually see the opponent strategy by looking at swing analysis (like when hitters are looking at low pitches consistently when I use many low pitches, etc.), which is kinda unrealistic; I should be probing this by how and where I pitch, not looking at artificial swing analysis.

Are there any tutorials or practice modes you feel that are needed or changed to gain a better understanding of the mechanic? Well any more elaborate documentations for the game in general would be welcome. The manual with the DVD is way too cursory.

Misc: Not at the moment...

Analog Mode: Pitching

Biggest obstacle when using this particular mode:

(1) It's not an obstacle but it's far too easy to locate pitches.

(2) It's rather hard to miss widely. Vertical location is fine, but I hardly ever miss widely outside or inside.

Any improvements to the mechanic that could help overcome the above obstacle: Pitcher control/consistency needs to be worse in general. At the same time, the degree to which these get worse should be affected more by BB/9 and individual pitch control ratings. I feel it is not pronounced enough for different pitchers as it is.... And overall, this should be tuned to produce roughly 3 - 4 walks per 9 inn... It'd be hard to balance everything with different user skills involved, but.... YOU CAN DO IT!!!!!

Maybe doing above will take care of (1) as well as (2)... not sure if any drastic interface change is needed, but making control worse is definitely desired. Walks are part of the game... Unfortunately I haven't spent much time how interfaces could be improved.

Anything you feel is missing or needs to be taken away from the mechanic to make it function better:

If applicable, is the visual feedback/graphics ample when using this mechanic. If not what is lacking or needs to be changed?

Are there any tutorials or practice modes you feel that are needed or changed to gain a better understanding of the mechanic? Ditto.

Misc:

Quote: Originally Posted by **stealyerface**

> *Analog Pitching: How about just scrapping the L-Stick pre-pitch aiming mechanism all together, and using the R-stick solely for the pitching process.*

I first thought this would be a cool idea to reduce pitch accuracy, but then I realized that it's already the R3 that really determines all aspect of pitch location, regardless of where we place the target with L3.

The thing that differs in vertical and lateral locations is that, vertically, you always want R3 to move the same way to have a decent release since L3 determines the pitch height, and a perfect release means the pitch will go there in terms of height. But lateral pitch location is pretty much entirely depends on how you move R3; don't know if there's any penalty for missing the initial target laterally, but it really doesn't matter where the initial target was as far as where the pitch really goes laterally. And probably the biggest reason why we rarely miss wide, as already pointed out, is that the strike zone is quite wide on the pitch meter. So obviously one way to reduce the lateral pitch accuracy is to reduce that width...

But I actually think **no visual aids whatsoever** for analog pitching is an idea worth pursuing. (Obviously I'm not talking about having this as an only option for analog pitching. Just as we have timing and zone for button hitting, we can have multiple interfaces.)

I think there are different ways to implement this, but one way that's pretty close to what we have already is just using R3 to control both the release point and the lateral location. The release point needs to be a bit earlier to throw a high pitch, a bit later to throw a low pitch. You pull up right or left to pitch inside (to RHB) or outside. Or we can add "zone pitching," which uses L3 to determines the (visually hidden) target and use R3 for release timing.

Do you think this would be too hard? Release timing with respect to pitcher delivery would be slightly different to different pitchers, but I actually think the learning curve would be fun, not painful. Pitching practice becomes more important obviously, but it might be enjoyable to find the right release points for all different pitchers on your team. For RTTS, you obviously need to know your pitcher very well...

One thing that I noticed is that people are quite good at making adjustments (at least myself because... ugh, I'm a genius....no). When the patch came out to reduce pitch accuracy, I initially thought it was great because I started walking people, something I did rarely pre-patch. But then after racking up several more games, I wasn't walking anyone again. That's because I made enough adjustment to make myself a good control pitcher, despite the patch made it slightly harder.

So even if pitching with only a few visual aids sounds too difficult, it may actually work out. Basically zone hitters do this already.

And one reason why I kinda like this idea is that I really don't like pitch meter interface in general. Not that it's a bad interface (in fact it's good and many games use it for good reasons...), but when I pitch with meter as a visual aid, the game becomes just that (i.e., time things right to hit the visual mark) and I don't feel like pitching baseball. When I pitch, I want to be looking at the catcher's mitt, not an imaginary visual aid.

For a higher difficulty level, I think this may add a cool interface.....

Analog Mode: Fielding

Biggest obstacle when using this particular mode: Not that there's anything wrong with the current system, but I feel the fielding and base running are parts of the game that can get spiced up a bit some way or another... I think part of the issue is that these two parts of the game uses user inputs as "commands" (via preloading) rather than directly dictating actions on the field. This is pretty different from pitching and hitting, where we have a lot of direct controls on how players move.

Any improvements to the mechanic that could help overcome the above obstacle:

Don't know if this should go here as analog suggestions or general gameplay suggestions, but...

(1) Some user input for "catching" the ball. Currently many parts of fielding is automatic or AI controlled. Making an error for the most part appears to be based on a dice roll. What about making user do something, like press a button, when catching the ball? If the time is way off, it would increase the chance of committing an error, for example.

(2) Momentum -- this is fairly good I think... I enjoy more I learn how to control the player momentum. However, I think the current momentum system kicks in only when user input goes almost opposite to the direction in which the player is currently moving. So when I "circle" L3 so that a player goes in round unphysically (many kids online make this kind of funky moves to kill time on the field haha), there's not much resistance in doing so. Similarly, I can almost suddenly make a perpendicular move, which is unphysical... the player should keep moving in the original direction a bit...

I don't wanna go in detail about physics but the devs can improve the momentum system by considering the momentum in terms of perpendicular vectors, one in the forward direction with respect to the player and another one perpendicular to it. As long as the L3 is not pushed in exactly the same direction as the player movement, there should be momentum left toward the.original direction. I think that will make the momentum system even more realistic...

Anything you feel is missing or needs to be taken away from the mechanic to make it function better:

If applicable, is the visual feedback/graphics ample when using this mechanic. If not what is lacking or needs to be changed?

Are there any tutorials or practice modes you feel that are needed or changed to gain a better understanding of the mechanic? Oh fielding really would benefit from good documentations and tutorials given the not-so-apparent complexity that we have to discover by playing a lot. There was a major discussion on the slow animations thread earlier in the year...

Analog Mode: Hitting, Pitching and Fielding

Biggest obstacle when using this particular mode:

This isn't specific to any mode, but one (potentially personal) issue that I have with analog R3 stick control is that I have much greater trouble toggling it precise to aim left than right. This is likely because the natural movement of thumb when holding the controller in a standard manner is from bottom left to top right (going from 8 to 2 in clock). To be clearer, I have much better command on inside (to right-handed batter) than outside pitches when controlling a pitcher using broadcast camera view.

Any improvements to the mechanic that could help overcome the above obstacle:

I have experimented with things like slightly rotating the controller so that my natural thumb movement becomes the 6 to 12 motion with R3. While this can work well, it's not very comfortable and access to other buttons become awkward.

I don't know if this is any better, but one thing that could help is to allow gamers to calibrate R3 (perhaps before gameplay) like with Move, so that we can define our own "6 to 12" direction with R3. Then I'd use the natural movement of my thumb to define that direction, and just move slightly off of it to pitch to inside/outside.

Maybe I'm just not finding the best way to use R3 for myself, but when I hold the controller normally, pitching to outside (against RHB) is much less reliable than inside. I wonder if others have similar issues. This obviously is an issue with throwing (in fielding) as well.

Road To The Show" Suggestions, Improvements or Tweaks for MLB 12

RTTS Gameplay

Base running

-- **Add player momentum when stealing and getting picked off**. When I get on base, I use the R1 (lean) + L3 to left trick to avoid constantly getting picked off on one-step lead. What I noticed is that when I then take off from 1B and the pitcher does pick off to the 1B, I can make a sudden move back to 1B and be safe. This probably is just a bug of something looked over...

-- **Pick off play to 2B need to be quicker**. I can often steal 3B with an aggressive lead when the pitcher does a pick-off move. Don't know who is the bottleneck (pitcher or a player covering 2B), but this happens a bit too often.

-- **Make it easier to steal**. I've played quite a few games with a player with max BR attribute ratings (as well as the speed), but I still find stealing a bit too risky unless the pitcher has a very slow delivery and/or throws an off-speed pitch. I'm not sure if this aspect of the game is entirely based on physics and timing that actually happens in the field (as opposed to some adjustments done based on attribute ratings, to make stats in line with reality; my guess is this is what's happening in the game), but even in the RTTS BR practice, it is very very difficult to steal a base unless the jump is near perfect. I doubt it's that hard in reality.

-- **Acceleration**? This probably belongs to a new idea category and not a tweak, but what about making L3 stick sensitive to how much you want to accelerate the base runner? Deeper you toggle, faster you want to accelerate. Currently it's either go or stop.

Misc.

-- **Add pitcher/batter analysis for last 10 (or so) games**. I like looking at those analysis screen to understand what my strengths/weaknesses are, and if we can analyze this over a few games and not just one game, it would be even more useful. I think the current analysis screen in Locker Room is quite an improvement over MLB 10, but it can even be better.

-- **Signs should be given at a more reasonable time, not just with the first pitch of an at bat**.

RTTS player progression

-- **Abolish achievement goals entirely and make player movement performance and need based**. Player movements within an organization become unrealistic more often than not (see other posts), because of an RTTS player having to achieve all these goals, especially with those involving attribute ratings. Now the real culprit perhaps is how strong attribute ratings affect GMs' decisions, regardless of

what the real performance/stats say. So, even when you are overachieving (with respect to ratings), the player might not advance, vice versa. To me improving ratings following those goals is unnecessary chore and only makes RTTS unrealistic.

Since most of my RTTS players end up being brought up to the MLB level as part-time players (which lead to less opportunities to earn training points), I often deliberately do not achieve those goals to stay at a lower level where I can play as a starter. This shouldn't be necessary... if the organization think a player is a starter material, he should be given a time to develop as a starter. So all this really leads to a need for a more realistic team management A.I... but meanwhile I don't really think achievement goals should have such a strong influence on how the player gets used by the organization.

Also, all those achievement goals get in the way when I want to just create an RTTS player with deliberate caps in his abilities. If I play normally, within 4 or 5 years I end up maxing out all the attribute ratings and he becomes a 5-tool superstar. While it can be fun, it's also fun to play as a player who plays to his role. The RTTS mode appears to have its own notion of how the player should progress, and if I don't use training points to those attributes, he gets demoted for no reason other than not achieving those artificial goals.

Overall, it's good to have a way to let us know what we should train, but that shouldn't dictate how the player gets managed within an organization.

-- **More meaningful attribute progression curves**. The introduction of "player types" was a good step forward. However, it only gives slight offsets in attribute ratings that matters just at the beginning of a career, and within a year or so the type differences can be washed out by adding training points. While this is okay in a sense that we have so much freedom in developing players to our likings, it just makes the mode so character-less... we *know* we end up with a 5-tool superstar with similar abilities.

Now I'm not very familiar yet with how the attribute ratings change (in the franchise mode) according to "player potentials" but if that's a sort of mechanism to "cap" some ratings so that all players have different strengths and weaknesses, then it might make sense to carry over that type of "cap" system so that it becomes easier/harder to develop a player into a certain type.

For example, if you create a player as a speedy, contact hitter with little power, it would be easier to add points to Contact and Speed ratings, but not Power. Say if you put 40 training points to Contact, then Contact would go up by 10 but if you put the same training points to Power it would go up by 4... just an example.

I just really enjoy playing a player to his style, so it might not go well with gamers who just want to dominate and feel triumphant... but RTTS can get really dull after maxing out all attributes.

-- **Don't tie player types into positions**. Related to above, I don't think it's necessary to tie player types to fielding positions. A slow power hitter like Prince Fielder *can* play SS, just not very well suited. But that decision doesn't need to happen at a player creation stage.

MLB 12 - Online / Online League Suggestions

Online Game Lobby & Play Now

There should be **separate Play Now options for guess pitch on and off**. I never play with guess pitch on and having to wait two weeks till the options swap is kinda lame...

Or maybe **a better generic filtering system for choosing potential opponents** would do as well? Like you have a range of options that you accept for opponent (like guess pitch on/off, hitting/pitching difficulty, strike zone on/off), and when an opponent is selected randomly, you only face the gamer that satisfied the conditions. If no matching gamers are found, the prompt would say, relax the restriction, etc...

Remove the lobby's 32 gamers limitation.

It might be nice to be able to **create our own lobbies**... Lobby is good because you can look for random players who wish to play games certain ways. Yes, we have online leagues, but having to set it up is quite a chore and counting on same users to play repeatedly most often doesn't work (so many guys just are inactive or quit for no reason). With lobby, the ones with more popular setups will be frequented by more users and become more active. You can find random players all the time, and can meet new players easily. With current lobbies, options are limited and someone as old as dinosaurs like myself who wants to play a simulated MLB game ends up playing 12 year old kid who just plays like he's having a boner and quits once he starts losing... that is not fun. If I can set up a lobby for "simulated ballgame" or something, then we don't have to have that frustration. Also, **within a custom lobby, the lobby "owner" should be able to adjust difficulty level, roster, sliders, etc. Basically it's online leagues but with a less rigid format and allowing any users to participate.**

Better gamer info. The current sportsmanship rating is very useless. We have enough info to see if a player is a cordial gamer or not (DNF%, quits, DC, SB/CS, etc.), but it would be nice if there is a better metric to do this quickly, rather than flipping the gamer card...

Online League

The "challenge fail" bug must be fixed. This one is simply inexcusable...

I wish the online league is **seamlessly integrated with offline franchise**. As it is, online and offline leagues are entirely different entities, and I feel the online league is not very sophisticated... it just looks like a way to set up a bunch of exhibition games between the same players over and over. Absolutely no fanfare on winning a league.

Allow games against CPU to be incorporated into league results? Online gamers don't take thing seriously enough to complete whole season, so it's almost expected that some teams won't have gamers to take over. Why not let us play against CPU and use the results to update the league results? The current sim option is fine for saving time, but then we have not gamer input on it.....

Online Gameplay

I'm sure every effort has been put to reduce lag, so I won't go there... I find online gameplay to be quite good in general, especially with better internet conneciton. Not as good as offline of course, but good enough for the most part.

On the other hand, the **connection meter is useless**. It's either green or red, and doesn't tell anything about how good connection really is. For me, the best way to tell how good a connection is is to use pitching meter, and how early I have to react to hit the "bar." When the connection is slow, I have to react earlier, when fast, slower. Maybe there can be a way to quantify this using ping info or something? That would also help reduce people quiting mid game due to poor connection, because we will know visibly when the connection is bad... So earlier suggestion for pre-game pitching/batting practice probably is a good one; give players an option to quit right after the practice. The remove the 1st inning quit not counting toward official stats/record.

Slow animation issues are the biggest headache. This wouldn't surface against CPU, but humans are quick to find glitches and exploit them. These things maybe hard to find unless many HUM vs HUM games are played specifically to find these exploits but fortunately many glitches have been mentioned already, for example: the slow movement of the first baseman allowing a runner to advance an extra base on a bunt or routine infield grounder; bunt successful too often because of slow infielders; early steal too often successful because of slow pick off; stealing home on runners on 1B and 3B might be too easy upon 1B stealing 2B due to slow motion by 2B/SS.

Don't know what the best way is, but a **better mechanism to deter people from quitting** so easily would be nice. Not that quitting is wrong for good reason (like emergency, has to crap on a sudden urge due to diarrhea, etc.), but you quiting for no good reason is the opponent's wasted time. Let quitters take severe consequences (it's game anyways... who really cares about XP points and such... just dock off A LOT of points; they can always keep making new accounts.....just make it a bit harder to waste others' valuable time...).

Forfeit by umpire. Fortunately I've never been a victim but some online players intentionally walk or hit batters after batters with an intention of inducing the opponent to quit by prolonging game time. This is really a bad taste and bad manner which can be prevented by implementing a forfeited by umpire, when an unusually high number of these event happen.

Allow slider adjustments for each exhibition game. MLB 11 online is heavily pitching dominated. Yes, we can use lower difficulty levels, but I often felt something like lowering pitch speed or using more lenient HUM timing would boost offense to a more enjoyable level.

Make check swing detection more lenient. Most likely due to reduced FPS, it feels that check swing is much harder to do online than offline. This being my first year playing the Show online, I'm not sure how things were before, but the amount of strikeouts you see online is simply unrealistic, even if you take into account the impatience of most gamers. So if necessary, an adjustment should be made such that it becomes more lenient for hitters.

Gameplay/Presentation for MLB12

Gameplay

Fielding

-- **Too few wild pitches**. It was overcorrected by the patch for '10. Pitches in the dirt should result in more errant plays. Currently catchers are too good keeping them in front. On a related note, **dropped third strikes should result more in catchers tagging the batter runner**, rather than throwing to 1B for completing an out. If a gamer is in control of a hitter, then I think it would make sense that the batter doesn't break to 1B on a dropped third strike, unless the gamer instructs to do so.

-- CPU-controlled **fielders, especially outfielders, make too many unrealistically stupid errors**. Forcing them to make errors is perhaps necessary to produce errors in the amount comparable to real life, but when I was watching CPU vs CPU games, I noticed a vast majority of outfield errors are of "totally botched" types. Like dropping very routine fly balls (rare IRL), throwing wide to cutoff men (also rare). If they make more realistic errors (perhaps on harder plays), that would be great...

-- **Too many outs at third base**. Don't know if this is the result of overly aggressive base runners or too accurate long throws, but in general, there are too many outs made at third base. This can be balanced easily by long throws less accurate and/or making base runners less aggressive.

-- **Making reaction time worse for pitchers and catchers**. I think pitchers handle too many (hard-hit) infield grounders. Also, catchers sometimes have AMAZING reactions for batted balls around the home plate, catching them as soon as they were hit. I think these guys need to have longer reaction times. Pop-ups should be harder for catchers to handle so it's also good for that...

-- **Make collision detection between pitcher and batted balls less lenient**. Way too many hits to center end up hitting pitchers. They should simply go through more.

-- **Make fielders ready to make the next action quicker when completing a putout**. The biggest "exploit" of slow animation happens when a runner takes an extra base on a routine grounder or bunt, because the first baseman is slow to get off the bag and be ready to throw to the next base if the runner tries to sneak an extra base in. This is also needed for double steal with runners on 1st and 3rd.

-- **Make fielders quicker on handling bunt**. Enough said.

-- **Make fielders make roughly right proportions of throwing and fielding errors**. From CPU vs CPU games, we found that throwing errors were way more frequent than fielding errors. These have different effects on games so they should be generated by roughly correct proportion compared to real life.

-- **Fielder speed and arm strengths**. Don't know how these are tweaked, but I think it goes a long way to adjust these using real life numbers. I didn't time things myself, but I had impressions that in '11 fielder speed was too fast and arm strength too strong at default sliders.

-- **More variety in errant throws**. I think there are only a few ways that throws go wide off targets. For example, some times, an outfielder can throw waaaay over the head of third baseman or catcher IRL. That doesn't happen in the game right now. One thing that can be added is to **increase the**

probability of error on a very hard throw from a short distance. It is simply harder to handle; it may be fun to think about *not* throwing too hard to be considerate to the receiver... takes some skill.

-- **Less accurate long throws (to 3B).** I think the reason why there are too many outfield assists (to 3B especially) in the game is that long throws are often too accurate. In the case of outs at 3B, throws are often too accurate even though they are long, and the third baseman can be very very quick (unphysically) to tag the runner.

-- **Fielder momentum**. I also made some comments about the momentum system in the "analog control" thread

Analog control suggestions, improvements or tweaks for MLB 12

so I'd appreciate if you take a look...

-- **Cutoff man location**. Often when the first baseman is the cutoff man, he's very slow to be near the pitcher's mound and the cutoff throw goes to him standing in a weird location near 1B; the first baseman should be quicker to be in his right cutoff position.

-- **Throws to the base, not to the fielder**. In the game the throws are made to the player covering the base, not to the base. That's reasonable if it's a toss and the fielder still has some distance to walk/run before he reaches the base, but in general, if the player will be on time to cover the base, the thrower would throw to the base, not to the fielder covering the base. The same goes for cutoff men.

-- **More random paths by outfielders getting to the ball**. The current systems appears to compute the shortest distance between the fielder and the ball landing location, and the moving fielder takes that shortest straight path, effectively handling it perfectly every time. Many have mentioned this in the context of outfielders handling balls off the outfield walls perfectly every time, which reduces the variety of plays leading to extra base hits. I have read somewhere that the devs have been trying to make it less perfect but haven't been able to do it reasonably well. Here are a couple of ideas...

(1) **Let each fielder start moving in a random direction slightly off the best straight path**. (Yes I'm aware this does happen sometimes with fielders with low reaction in the current system.) This can be tied in with fielder reaction to some extent. A fielder with the best reaction takes close to the best path almost every time, but one with a lower rating can be off that path a lot of times. As the fielder continues to run, he continually corrects his path so that eventually he figures the shortest path to the ball. And (2) when the fielder really isn't surely on the best path yet, the **fielder should be running at a slower speed**. Once he knows he's "on track", he accelerates to his max speed. IRL, you see a lot of times the outfielders are not really running at full speed when he's tentative (unless he already knows that the landing location is almost far out of his reach, in which case he should be running at full speed even slightly off the best path).

Again, if this sort of thing can be added with a more detailed fielder momentum system, players would really look organic out in the field. Some fun stuff to improve there!!

Gameplay

Misc.

-- **More hit variety**. I think there can be more variety in batted ball. This would make fielding more fun and less routine, particularly important for RTTS fielding. Especially lacking, I feel, are high bouncer in infield and blooper in outfield. There should be less of weak hits (both popups and grounders) near the home plate; 90 mph pitches coming off the bat shouldn't be hit that weakly, otherwise the bat should be shatter to absorb the energy....

-- **Toned down wind effects**. Was way too strong in '11.

-- **Revisit rain day algorithm**. I hardly ever saw rainy days in Tacoma.

-- **Sunny day games might be too bright**. Some have complained that the stadium/grass is too bright in day games, making it hard to see the ball in the field.

-- **Custom camera views in online gameplay**.

-- **Player eye view**. This might be a maniac category, but it might be interesting to add a camera view that follows that view of the player you are controlling. For RTTS this makes sense. You just put the camera at the eye of the player, rather than behind him a bit. It's just a fantasy of actually being that player. Whenever that would make much more sense to have a wider view, the view swaps between the player view and a wider view like implemented already.

-- **Less ground balls, slightly more line drives/fly balls**. According to the stats that I have accumulated in CPU vs CPU games, the game produces about 5% more ground balls than real life. That should become fly balls and line drives. (Although I have to say the game's identifications of those batted balls, which I just copied from the batter analysis screen, may not reflect how really the balls travel...)

-- **More reasonable pitch speed on a pitch out**. Every pitcher throws a mid- to high-heat on a pitch out. Instead it should be a similar speed with the fastest pitch the pitcher can throw.

– Also, **boxscore and game log should be available for every single game**. I noticed that for some games (like all-star games) they are not accessible from the schedule screen.

This is a stretch but retrosheet type of game log would also be a cool addtion....

Gameplay

Batting

-- **Either make analog/button response or swing speed faster**. Many have mentioned it's too hard to pull inside high fastball this year. I definitely feel it's too hard with analog. With button it's easier but many button users have said the same thing.

-- **Less pronounced timing differences between inside/outside and high/low pitch**. May actually be debatable if this is really desired... I do really like how such differences exist (and the devs know enough to realize this in the game). What I mean is that I feel kinda weird that many gamers end up hitting away inside pitches and pulling outside pitches more often than I actually think they happen IRL. The reason why this weirdness happen, I believe, is that the gamer tends to hit *everything* with similar timing given that all they do is to press a button/push up stick (as opposed to IRL you need different swings to hit inside/outside pitches). So if an inside pitch is hit with the same timing with an outside pitch, then the former tends to be hit away and the latter gets pulled (at least the timing right now is done in such a way I believe...). I found this to be the case, at least in '11. I feel the difference is a bit exaggerated now. Without knowing how the hit timing is coded I cannot really be to the point, but some adjustments in either timing window and/or the optimal timing for hitting right to the center would help... don't really know what gives me that weird feeling about too often pulling outside pitch. (On the other hand, going away with inside pitch might be because of the issue above, swing speed/response being slow....)

-- **Tone down opposite field power**. Not many hitters hit for power to the opposite field; in fact only 10 - 15% of HRs goes to the opposite field, e.g., http://www.hardballtimes.com/main/ar...the-other-way/. I think this would have a cool effect of the gamer trying to pull (which leads to reacting/swinging earlier, which leads to more whiffs, something power hitters do often) with power hitters and add a dimension to hitting strategy.

-- **Make it easier to make contact with bunt**. Bobhead's earlier post is right on there.

-- **Make it easier to pull bunt attempt**. It's too hard in '11.

-- **Add offset view(s) with opposite perspective**. This was actively discussed earlier in the forum. All preset offset views right now are shifted such that outside comes straight toward us the gamer. This makes it easier to have good eyes on outside pitches. In reality it probably should be opposite, i.e., outside pitches should be harder to judge. So some offset views should be added to accommodate this. The challenge is that with some hitters who stand close to the plate, it may become harder to see the release point of the pitcher (also when bunting). I love offset views so I'd really appreciate some thoughts here.

-- **Allow even more flexible custom camera placement**. I have this fantasy on how I want to hit with almost the same view as myself standing in a batter's box, like bringing the camera on the same eye level as the hitter. Currently the customizing camera doesn't allow such view; the camera has to stay behind the hitter. Would it be possible to make it more flexible? Not a big deal but...

-- **Exaggerate the change in ball size as the pitch comes to the plate**. Our brain judges how fast a ball is coming toward us by how much the size of the ball visually changes. Since we can only see 2D projected ball, it's very hard to distinguish fastball and change-up in the game. This is made worse by

the fact that we cannot pick up the seam in game. So, in order to aid us by having more visual cue, why not add a mode (a mode because making things optional is always better than forcing us to play in certain ways...) in which the change of the ball size is a bit more exaggerated? I bet this would make it easier to time pitches. After playing a few hundred online games, I noticed only the very very best gamers can distinguish fastball and change-up. The rest simply cannot. So why not help the rest? It's better than guess pitch anyways...

-- **(Only online gameplay) Make check swing easier for online play**. Offline, check swing is implemented very well. Online, I think it's definitely much harder, and I presume that's because we don't have as much resolution (in terms of FPS?). Given that there are so many strikeouts in online games, it may help to make check swing easier just for online play.

Gameplay

Pitching

-- **Pitcher control should be worse**. Not reproduce discussions here cuz there are a bunch already.

-- **Change-up (or other pitches whose pitch speed become slower as the speed rating goes higher) speed should be varied off the fastest pitch a particular pitcher throws**. Don't know if it's easy to do, but when a non-pitcher is on the mound, he has only fastball and change-up, but his change is actually faster (in 80s) compared to his fastball (in 70s). This is weird. It seems changeup speed is hard coded against the pitch speed rating, but it would be cooler if the speed rating is tied to how much the pitcher takes off from the fastest pitch he throws. After all that's what change is for...

-- **Make knuckler dance less**. It dances way too much and unphysically.

-- **More hit-by-pitches**. CPU pitches rarely hit batters in '11. There should be more to bring the stats in line with MLB.

-- (For simulation engine) **Very high BB/9 attribute rating should result in better BB/9 stats**. What I noticed by looking at game-generate stats vs. attribute rating is that very high BB/9 attribute rating doesn't really lead to very very good control pitcher who rarely walks batters. For example

Calibrating ratings to game stats

according to my OCD attempt to calibrate BB/9 player rating to BB/9 stats, even the best BB/9 rating leads to pitchers who walk about 2 - 2.5 walks per 9 innings. Best control pitchers in MLB on the other hand can have about 1 walk per 9 innings (e.g., Roy Halladay, Cliff Lee). So more dynamic range here would be great to reproduce their performance.

Gameplay

Baserunning

-- **Remove preloading steals**. This may be too drastic a change, but what I mean is like what Bobhead said earlier. Make user input count for how good a jump a base stealer has. I'm just taking it further and make it like "early-steal" which is already implemented in the game. If you think about it, there really isn't anything that clearly distinguishes early-steal from regular steal. IRL a runner can take off whenever he wants. And as much as I hate online early stealers who do it just to annoy the opponent, I actually like the dimension it adds to the pitching, the fact that I have to hold runners. With the chance of early-stealing, I actually need to think about when the runner takes off (with regular stealing, I don't have to because it will always be after I deliver to the plate). And thinking about runners, it takes off concentrations toward hitters. That's exactly what running game can add to the game... and I like it! So why not make it the right way to steal? If this is implemented, then it's also essential to make it very hard to go back to the base; once a runner takes off, let him run for a while till giving an option to go back to the original base. What makes early stealing annoying online is that a runner can keep going back to the base easily... it just prolong the game time.

-- **CPU early steals**. So the previous item explains why I think running game would add a depth that's currently lacking. Perhaps a significant change in the BR interface won't happen for the '12 installment, but to make HUM players guard against running games against CPU, it may be a good idea to add CPU early steals. That way, at least HUM should be aware that CPU base runners *can* take off if you don't pay enough attention to them.

-- Given a major change in base running/stealing interface probably won't happen, **give less jump with an early steal and/or make pitchers and fielders react faster**. This is mostly online gaming issue, but many gamers use early-steal (with a regular steal timing) just to get GREAT jump on steal. It doesn't matter who the stealer is, David Ortiz steals like Rickey Henderson given an early steal sign. Prince Fielder is a second coming of Lou Brock. This makes home steal unusually successful.

-- **Tone down pickoff success**. It's just ridiculously easy to pick off runners once they take just one-step lead.

-- (for RTTS) **Make it easier to steal**. To me it's too hard in '11 to steal even with BR speed/ability/aggressiveness maxed out.

-- (for RTTS) **Forward by a pitch**. I like controlling base runner in RTTS but it takes too long waiting for every pitch. 1/2 or 3/4 at bats is only a limited time-saver, but when I'm trying to steal, I actually want to wait and run on a particular count, which is only possible if I go through every pitch. So if we can skip forward pitch by pitch, it would be cool...

Presentation

-- **Make fast play mode even faster**. There may be places (like foul ball, between pitches) where the time can still be saved. I like watching CPU vs CPU games, but taking 45 min per game on average is still kinda long.

-- **Super fast play mode**!! This belongs to a wishlist category, so I'm not really hopeful about it, but if possible I want all my games (those played by myself and between CPUs) to be played through the same simulation engine. Simmed games go through a different engine and just knowing that takes a certain degree of realism off of playing through a season, not just because I don't like the feeling of "well it's all dice rolling after all" but because I think the gameplay engine is very well done. I'd love to use that mode to "sim" all the games that I don't play if possible, even if it takes a min to sim a single game...

Gameplay AI/engine

-- **CPU batters should do check swing/react more often**. Playing against HUM, check swing and that twitching happen quite often. Against CPU, it only happens a few times a game at most. These reactions should be used more often when CPU batters are fooled and/or have an intention to commit but decide not to.

-- **CPU batters should be fooled more often by timing**. I feel CPU batters are a bit too good at adjusting their timings. If I keep throwing fastballs 10 - 20 pitches in a row, he shouldn't suddenly be able to adjust to an off-speed pitch because (1) he shouldn't be sitting on it and (2) he should be committing with fastball timing; CPU most often can, however. The way CPU batters get (un)fooled by timing still feels a bit too random and inorganic to me.

This can really manifest itself against a pitcher with very low-speed change-up, such as James McDonald in the default SCEA roster. McDonald against HUM is a deadly nasty pitcher, since there's a huge speed difference (actually not realistic... this is a player editing issue as well) between his fastball and change. But CPU doesn't get fooled by his change like HUM does.

-- **CPU batters should chase pitches a bit more often**. While it has improved over MLB 10, I feel most CPU batters are still a bit too good at laying off pitches *just* off the black, especially with breaking pitches (on the other hand with fastballs it may be okay as is...).

I have not played with all different combinations of pitches of course, so not sure how much I can generalize this point... but I have quite some experience with a pitcher with fastball/splitter/forkball combo (think Clemens... though what I'm simulating really is Nomo). It's a classic power pitcher type who can get many Ks and/or ground balls with splitters that drop just off the strike zone once hitters get behind. Amazingly, most CPU hitters can lay off those supposedly nasty low pitches, only swing at them within the strike zone; they don't even check swing or twitch. Knowing the pitcher has a splitter, the batter shouldn't be committing to all of those splitters of course, but he should get fooled quite a few times when the pitch location is near perfect like that. To me it is as if the CPU batters aren't really aware of the pitch movement going from within the zone to just outside. Unless the CPU is deciding to lay off all low pitches for some reason (which I don't think is the case because he would commit to fastballs low in the zone), he should get fooled more by nasty pitches like those.

-- **CPU batters should swing and miss a bit more often against well executed/located pitches**. According to the stats that I have accumulated:

Stats-based sliders for CPU

CPU batters are slightly better at making contact than the average MLB hitters (18.1% vs 19.3% swing & miss rate with sliders at default). Now I actually think the number is good and comes close to real life and I could make CPU swing and miss slightly more by reducing Foul Frequency slider, bringing it slightly closer to the MLB average.

However, CPU hitters might still be slightly better at putting balls in play than real life. Unfortunately I haven't accumulated relevant data to back this up, but you see a couple people in this thread talk about the impression.

Now, I like simulating strikeout pitchers so I created two this year in RTTS: David Cone (a power

pitcher with mid-90s fastball and a wide array of really nasty breaking pitches) and Hideo Nomo (two pitch pitcher with 90 mph fastball with nasty forkball/split). Both were great strikeout pitchers (K/9 > 9) at their primes. My strategy was maxing out their K/9 ratings as well as individual pitch break ratings, thinking that was the best way to increase swing throughs. and the result is... I can barely record better than average K/9 stats using them (I have played with AS/HOF/LEG)!! What happens, I guess, is that CPU batters put balls in play more often than MLB players against this type of pitchers IRL. In general, the pitchers with best K/9 ratings barely crack K/9 > 9 in CPU vs CPU games.

My suggestion is **make batters swing & miss more according to K/9 and individual pitch break ratings**. I actually don't know what pitch break rating exactly does (other than the obvious fact that it makes a pitch visually breaks more), but in order to make the difference between strikeout artists and contact pitchers more pronounced, pitches with larger breaks (like curve/slider/forkball) should induce more swing and misses. According to this article:

http://bleacherreport.com/articles/7...011-mlb-season

MLB batters can swing & miss the best breaking pitches 30 - 50% of the time (roughly 15% for fastballs). That's A LOT and I don't think CPU does so this much in the game. Obviously such percentages are relevant for pitches properly set up (batters will handle if they know the pitch is coming)... such that a breaking pitch with high pitch break rating, when properly set up, ends up whiffing batters that often. For poorly rated/executed pitch, the % should be less.

The devs must be using pitchFX data to tune the game already (otherwise so many stats shouldn't be coming out very close to the MLB numbers already), but I hope they continue to do so for this sort of things... for individual pitches. If we have more dynamic range in what an individual pitch does, pitchers will have more pronounced characters, which is a bit lacking in the current game as pitchers with similar pitch selection can often really look similar in this game.

-- In general, each player attribute rating can have a slightly wider dynamic range in the result it induces. I'll elaborate on this later hopefully.... but basically what I mean is, when I eyeball all the individual player stats in a season that I simulated, playing all games as CPU vs CPU, I don't see very very good players performing much much better compared to the rest (and vice versa). On average as a league (something I religiously kept track in my simulation), the game reproduces the real-life stats very well. But there aren't as big variations in individual performances as their attribute ratings should imply. This is more so in pitchers than batters.

MLB '12 The Show- FRANCHISE-MANAGE ONLY MODE-EDITOR suggestions/improvements

Unfortunately I haven't been able to get to playing a full-blown franchise season myself yet... thinking I'd ran out of ways to enjoy the game and they haven't..... by the time I start one probably '12 will be out.

To really enjoy the franchise aspect, I'd really love to see improvements in stats areas. I enjoy accumulating and looking at historical stats, how they change over years, etc. I think that's part of being a fan of a particular franchise. Currently, we have few ways of reminiscing and looking back the history.... But that would be wishlist items and not called for here I guess.

One thing that could be improved within the current scheme is **the way all the numbers (ratings, stats) are accessed within the game**. Outside the gameplay mode, we have that player card thing which is accessible almost everywhere, which is nice (though perhaps you could make it more informative by not trying to make it look like a baseball card... could use more space). For individual players that is good enough but when you are managing the whole team, I'd like the interface to access the player info that I want to see to be consistent so that I can expect them to find where I know them to exist. Quite often, partial info like *some* player attributes/stats are shown, but not the rest; for example during the gameplay quick access menu, you can see power and contact easily but not others like PDisp/PVis. Quite often I forget how to access the info I want to see because interfaces are not uniform in different parts of the game.

Instead of scattering info like that, it would be more helpful if there is one (and only one) interface that we can access easily (maybe via hot bar) that list all relevant players, and see *all* stats and attributes for those players. This could be something based on the current "roster control" player list, but you can also access stats, attributes, etc., etc. Don't know if it's reasonably feasible without needing to do a lot of scrolling, but my main thing is having the once interface to access *all* player info. This would make it easier to know and manager players.

It would be nice if we can do filtering on the list via various criteria... teams, handedness, certain attribute value range, names, etc., etc. Something of a master player table like that would be nice...)

Also, some ways to **save retired player info** would be nice, maybe not now but if the career/year-by-year stats are already improved.

For CAPS, ways to record **numbers for some body/facial features, not just visual coordinate** would be nice. I wanted to transfer myself from MLB 10 to 11 in my RTTS and it was a chore to do because I can only do via reading off visual coordinates for some attributes.

That got me remember **roster/season import (from previous version of the Show)** would be very welcome if not mentioned already.

Lineup & Roster Handling

-- **Better lineup after fatigue/injuries**. I notice that when AI needs to take a player out of a starting line up (due to fatigue/injury), another player at the position simply replaces at the order where the player being replaced is assigned. This often leads to very weird line up quite often. AI should do re-ordering of line up if such a replacement is made, so that the final starting lineup looks reasonable according to hitters' abilities.