Which ITG Stepcharts are Bracket-Jumpiest?: In Which They Milk the 「A Boring Follow-Up Paper to "Which ITG Stepcharts are Turniest?" Titled, "Which ITG Stepcharts are Crossoveriest and/or Footswitchiest?"」 Series for All Its Worth in Publication Count After All, or: Hit Me With An Encore

Ben Blum

bblum@alumni.cmu.edu

Abstract

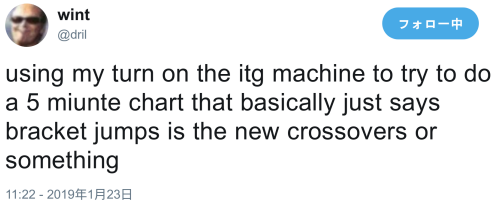
In which I break last last year's promise of no future work.

Categories and Subject Descriptors D.D.R. [*Exercise and Fitness*]: Arcade Dance Games

Keywords bracket, groove, in, jumps, the

1. Introduction

Recent work by (dril 2019) proposed the hypothesis that recent stepchart authors have grown bored with the array of technical ITG step patterns documented to date (Blum 2016, 2017), and have moved on to break the old model's one-foot-per-arrow assumption to allow for even more technical patterns yet. I paraphrase their main conclusion as follows:



All good researchers know that when rise the standards for software or hardware performance (or stepchart trickiness, as the case may be), they must revisit their own work to prove its ongoing relevance to the research (dance) community at large. Thus I must regrettably break the promise ~~I~~ the authors set forth in (Blum 2017) (see Figure 1) and revisit ~~my~~ their old future work section (XXX: they said to change any first-person "our prior work" stuff like this for double-blind review but like cmon this makes no sense? theyll totally see through this (TODO: maybe email the PC chair for advice? (FIXME: make sure to remove these comments before the camera-ready deadline!!))), extending it to handle these new so-called "bracket" jumps.

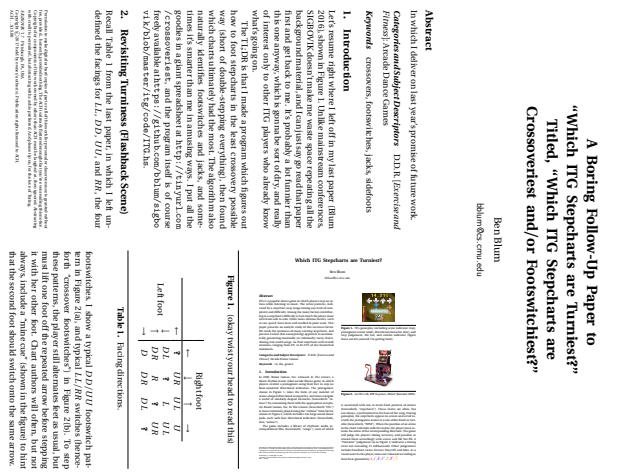


Figure 1. (okay i reused this joke from last time ok deal)

SIGBOVIK '19 Pittsburgh, PA, USA

ACH ...$13.37

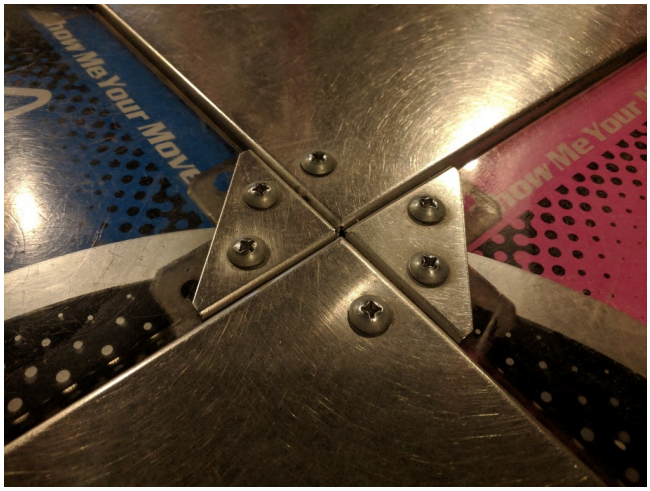


Figure 2. Detail of metal corner brackets, this paper's namesake. Photo credit JIM.

Figure 3. Down+right bracket-jump real-world example.

2. Overview

What, then, is a bracket-jump (which I shall *not*, henceforth, abbreviate for brevity)? Put simply, whenever two arrow-shaped obstacles proceed simultaneously towards the protagonist directional indicator targets (Blum 2016), while a novice player might think they must step with both feet at once, one for each arrow, experts often find it more convenient (i.e., less overall foot motion) to use whichever single foot is closer at the time to hit both arrows by triggering one arrow's sensor with the heel and the other with the little bitty toesies. Pads are typically constructed with a small triangular metal bracket at the corner of each arrow panel, as shown in Figure 2, which the bridge of the foot must cross to achieve this, hence the name "bracket jump". In case my prose explanation is not up to snuff, I also show in Figure 3 a high-quality graphics render of a player's typical foot positions during a down+right bracket-jump (henceforth "DR", et cetera).

The reader, or stepper, may notice the extreme angle of footing depicted in the latter figure, which is necessary to reliably trigger both pad sensors. Accordingly, this maneuver is comfortable (and hence preferable to a normal jump) only if the player is already facing in roughly the same direction (Blum 2016). Note also that the two "candle jumps", LR and DU, are not possible to bracket, unless your foot size is (physically) beyond the scope of this work. The next section will attempt to codify (ahem) when preceding patterns encourage the player to bracket rather than jump, and thereby identify how "bracket jumpy" each chart is.

3. Algorithm Design

I extended the crossoveriness et cetera algorithm from (Blum 2017) to reason about jumps, which previously it treated all identically, ignoring the arrows involved and allowing the player to reset her footing as desired. Now, it considers LD, LU, DR, and UR jumps as potentially bracketable, allowing the player to continue a stream of alternating feet uninterrupted therethrough.

Confession. When I was first brainstorming this project, I had some grand visions of unifying the turniness algorithm (Blum 2016)—which accounts for U/D steps to figure out how far the player must turn each step, but has no idea which way she is actually facing at any given time—with the crossoveriness one (cited just above; cmon how much do you want me to repeat these same two citations, gimme a break)—which totally ignores U/D steps and just figures out which arrows the left and right feet must each step—in some theoretically beautiful way to produce the unquestionably perfect footing sequence for each jump. However, as I was considering how to incorporate features from last time's algorithm, potentially allowing crossover brackets (Section 6.2) and footswitch brackets (Section 6.3), I realized that ultimately there would be no restrictions on what jumps were bracketable; the algorithm would oops simply twist and turn as much as necessary to bracket everything, and this paper would become just "Which ITG Stepcharts Have the Most Jumps?", and like who wants to read that.

One may think to simply try either bracketing each jump or not and seeing what combination gives the minimum turniness result, but since whether to bracket each jump or not affects subsequent steps' footing, each choice cannot in general be solved independently, and I hope you can see where this is going. Now, the last time I tried to solve an exponentially-sized problem, it took me 7 years (Blum 2018) (and I still ended up with a pile of heuristics after all anyway), so considering I started this project a week before the deadline, I opted instead to just code up

a bunch of ad-hoc rules to handle all the different bracket jump patterns that occurred to me to write test cases for.

As before, the code is available at https://github.com/bblum/sigbovik/blob/master/itg/code/ITG.hs. To give a sense of how much its elegance has been despoiled since the last version: 62 lines of Haskell (which computed crossovers, footswitches, jacks, doublesteps, *and* crossover footswitches) has grown to 156 lines (just to handle this one. new. feature), and the once-simple datatype definition of

```
data Step = L | D | U | R | Jump
```

has become the unwieldy:

```
data Arrow = L | D | U | R
data Jump = LD | LU | DR | UR | LR | DU | Other
data Step = A Arrow | J Jump
data Foot = LeftFoot | RightFoot
```

The general gist is that in addition to tracking which foot stepped the last arrow and optimizing for alternating feet, we also track which arrow(s) each foot was last on, and whenever we encounter a LD/LU/DR/UR jump, bracket it if both:

1. The last foot is opposite the foot required for the jump (e.g. L for LD), and
2. The last foot's last note(s) does not intersect the arrows involved in the jump (i.e., R not on D preceding LD).

In lieu of a detailed algorithm listing, I'll simply show some of those most notable test cases in pictures and explain in prose how a player would typically step them, which was pretty much my implementation strategy to begin with.

Figure 4 shows a bunch of patterns with a bracketable jump, and each subfigure in Figure 5 shows a corresponding unbracketable case. Whenever two jumps are pictured in the same unit test, we are concerned with whether the latter one is bracketable. I'll now explain how the algorithm handles each case.

a. In the OK case, the right foot can bracket every jump (condition 1), as the left foot is always on a different arrow than the jump (condition 2). In the NG case, condition 2 is violated for the second and fourth jumps.

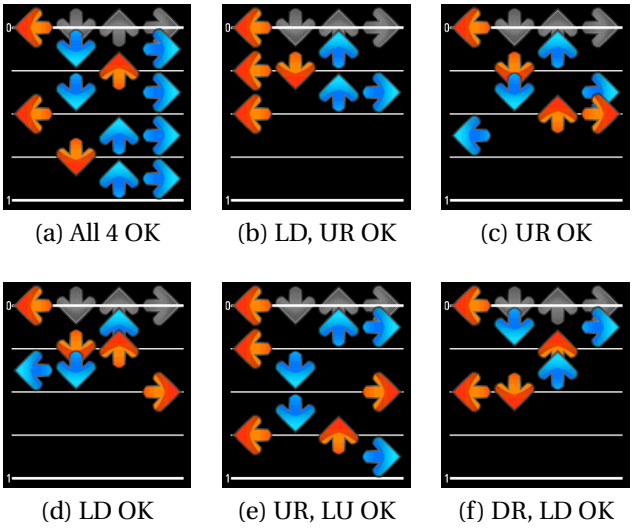
b. Version of (a) testing that preceding bracket-jumps also contribute to condition 2. The NG case would be a footswitch bracket requiring rather uncomfortable footing angles.

c. In OK, DR cannot be bracketed, but clears the prior footings so UR can. In NG, LR can be bracketed, meaning UR cannot while still alternating feet (condition 1).

d. DU jumps can never be bracketed, but can be stepped facing either direction, and this must respect prior footings. In both cases DU is stepped with right foot U, which allows bracketing LD but not LU thereafter.

e. In OK, the crossover does not interfere with either bracket and the player can alternate feet throughout. In NG, either doublestepping or jumping normally is required (or a future-work crossover bracket; see Section 6.2).

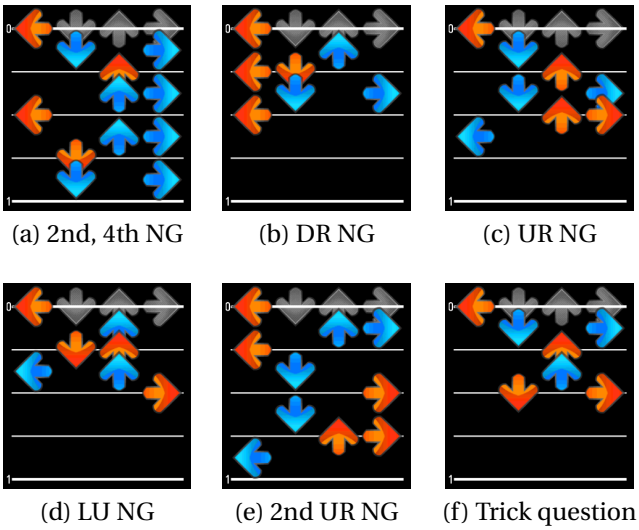
f. In OK, the player easily footswitches on U. In the corresponding case, the player can actually just jack U instead, stepping it twice with her left foot before bracketing DR again with her right.



(a) All 4 OK (b) LD, UR OK (c) UR OK

(d) LD OK (e) UR, LU OK (f) DR, LD OK

Figure 4. Examples of bracketable jumps.



(a) 2nd, 4th NG (b) DR NG (c) UR NG

(d) LU NG (e) 2nd UR NG (f) Trick question

Figure 5. Examples of unbracketable jumps, each corresponding to the similar pattern in Figure 4.

These examples are available as `test-bracket-*.sm` unit tests in the code repository. Hopefully armed with this understanding of the bracket jumping rules, let's now move on to real-world stepcharts.

4. Evaluation

The experimental corpus has grown even more since last time, now comprising 17340 stepcharts from 182 packs. Of note, all three Technical Showcase packs (collaborative packs where the community at large was encouraged to submit their freshest beats and judged on complexity) were released in the last two years, which have proved to be important sources of diverse bracket-jump patterns for this research. I would also have liked to re-rank the crossoveriest and footswitchiest charts from last time to include these submissions, but I frankly don't have time. The results spreadsheet of bracket jump counts (as well as more crossovers, footswitches, et cetera for charts newer than the last paper) is available for your browsing pleasure at `https://tinyurl.com/bracketiest`.

I pose two evaluation questions, to be answered in the following subsections respectively.

1. Which songs, packs, and/or step-cartographers are bracket-jumpiest?
2. Is bracket jumps the new crossovers or something?

4.1 Bracket-Jumpiness

I measured the overal bracketiness of each chart in two ways, first, by comparing the number of bracket jumps against the total number of steps, or the *bracket-jump step percentage* (BJS%); second, by comparing against the total number of jumps only, or the *bracket jump jump percentage* (BJJ%). The BJS% indicates a chart's overall density of steps requiring the player to step across the brackets (which depends on how jumpy the chart is to begin with), whereas the BJJ% measures perhaps the author's intentionality in patterning their jumps either as brackets, or as random where just some of them happen to be bracketable.

Table 1 shows the leaderboard for the former metric, and Table 2 the latter.[1] Note that the aforementioned recent Technical Showcase pack series put up three bracketiest charts in the BJJ% category; meanwhile, the UPS packs, known for their gimmicks and general lack of respect for player comfort, secure several top spots on the BJS% board. For comparison's sake, the overall BJJ% of the entire corpus is 19.8% (93k/470k), meaning that roughly 1 in 5 randomly-patterned jumps are by chance bracketable. The overall BJS% (a thoroughly less meaningful statistic) is 0.7%.

[1] When measuring BJJ% I excluded charts with fewer than 10 jumps in total, which put up a handful of false positives in the 100% ranks.

Ft.	Name	Pack	#BJ	BJS%
10	SOBA	Squeaky Beds &c.	366	83%
11	Get Off of My Way	UPS 3	115	28%
12	Hardware Store	Keyboard Coll. III	207	23%
13	Firestorm	BemaniBeats 3	153	31%
14	Bounce	UPS 2	126	21%
15	Ikaros Dynamite!!!!	UPS 2	238	34%
16	Mermaid Island	Tachyon Alpha	346	49%
17	Toccata & Fugue	CuoReNeRo M.P.	255	24%

Table 1. Charts of each difficulty with the highest bracket-jump density among all steps (BJS%).

Ft.	Name	Pack	#BJ	BJJ%
9	Drifting Away	UPS 4	26	88%
10	SOBA	Squeaky Beds &c.	366	99%
11	Save Miracles	ECFA 2019	42	95%
12	Electrical Paradise	Chic. Timing Auth.	35	100%
12	Encore	Tech. Showcase	29	100%
12	Nemeton	Subluminal	12	100%
13	Decadent Dandy	Tech. Showcase 3	12	100%
14	Nageki no Ki	Valex's M.4-A.A. 8	116	94%
15	Beach Party	Tech. Showcase 2	162	80%
16	Mermaid Island	Tachyon Alpha	346	79%
17	Zombie Sunset	Jummy Jawns 2	325	88%
69	koopa bling	UPS 4	39	98%

Table 2. Charts of each difficulty with the highest percentage of their total jumps bracketable (BJJ%). All participants of the 3-way tie for 12-footers are listed.

Next up is bracket jumpiness by pack and by author. Tables 3 and 4 show the top 10 packs in both bracketiness metrics, and Tables 5 and 6 the top 10 authors (filtered by having written at least 10 charts). For the packs, I also list the release year of each; note how despite the relatively even spread of years in BJS%, 2018 dominates the BJJ% leaderboard, suggesting that older charts' bracket-jumps arose by chance simply from having a lot of jumps to begin with. I'll come back to this point in Section 4.2. In honorable mentionth place is chart author Halogen–, whose 11 charts contain 154 jumps, *none* of which are bracketable. What a purist!

4.2 Historical Trends

I sought to prove (dril 2019)'s claim that bracket-jumps' popularity has skyrocketed in recent years of stepchart-making, by finding the progression of bracketiness across past years of stepchart packs. However, actually assigning a firm release date to every pack on my hard drive turned out to be a feat of internet archaeology unto itself, even involving archive.org for one step. I'll spare you the details, but you can peruse them in the second spreadsheet of the dataset linked at the start of this section. Ultimately, all packs older than 2013 had to be grouped together, as no

Pack name	Year	#charts	BJS%
Keyboard Collaboration III	≤2012	17	12.4%
Squeaky Beds and Leaky Faucets	2018	125	4.9%
Keyboard Collaboration I	≤2012	12	3.6%
CuoReNeRo MeGaPacK	N/A	460	3.3%
Mute Sims X2 WIP	2018	12	3.0%
r2112	2007	60	2.7%
Technical Showcase 3	2018	196	2.7%
FoxyMix 4 - Nuclear Overdrive	≤2012	67	2.6%
Technical Showcase 2	2017	80	2.6%
Gensokyo Midnight	≤2012	77	2.5%

Table 3. Song packs densest in bracket jumps.

Pack name	Year	#charts	BJJ%
Feelin' Rusty 3	2018	51	50.2%
Squeaky Beds and Leaky Faucets	2018	125	46.7%
TYLR's Technical Difficulties	2018	33	44.9%
Technical Showcase 3	2018	196	41.7%
Mute Sims X2 WIP	2018	12	40.9%
Jimmy Jawns 2	2015	55	39.7%
Technical Showcase 2	2017	80	38.2%
Keyboard Collaboration III	≤2012	17	38.1%
Chicago Timing Authority	2018	33	36.3%
DVogan's Tech Support 2	2018	32	36.1%

Table 4. Song packs with the most jumps bracketable.

pack collection website, facebook group, or laptop filesystem metadata could accurately date enough packs before then to give meaningful sample sizes for each year. I also decided to exclude tournament packs such as ECFA from this analysis, which curate existing stepcharts written possibly long ago.

Statistical significance. Stepcharts thus dated, I then summed the total steps, jumps, and brackets published in each year, and analyzed the resulting overall BJS% and BJJ% as a linear regression. Figure 6 shows the best-fit and 95% confidence intervals, plotted in R (R Core Team 2018) with ggplot (Wickham 2016). For BJS%, the ≤2012 bucket actually produced the highest data point (at 11.8%); I suspect this simply represents the fact that jumps were more popular overall during ITG's nascency (as corroborated by Table 3). Owing to this and also to its nature as an aggregation over nearly a decade (which spoils the linear model anyway), I chose to exclude it from BJS%, but kept it for BJJ%, which overall jump count has no bearing on. (Its linear fit has $\beta = 0.012\%$, CI = {-0.08,+0.10}, not significant.) For the two pictured distributions, their confidence intervals do not include 0, i.e., a flat line representing no growth, so I conclude bracket jumps' increased popularity is statistically significant.

Bias. This analysis is prone to selection bias in my own pack downloading habits: supposing I suddenly became more interested in playing brackety charts recently, that would certainly influence how many bracket-jumps appeared each year in this corpus. However, I believe the BJJ% measurement adequately compensates for this, for if I suddenly became interested in stamina instead of technical and added a bunch of 10-minute trance packs, that would impact only BJS%. As far as I know, I show no preference for more or less jumpy charts overall than the community average, leading me to trust the BJJ% test.

Author name	#charts	BJS%
Paul J Kim	45	9.9
sssmsm	45	4.4
Liam	11	3.4
Snooze	22	2.5
M. Emirzian	26	2.2
bblum	40	1.9
Ninevolt	10	1.7
B. Vergara	95	1.6
C. Emirzian	18	1.6
B. Dinh	10	1.6
Renard	51	1.5

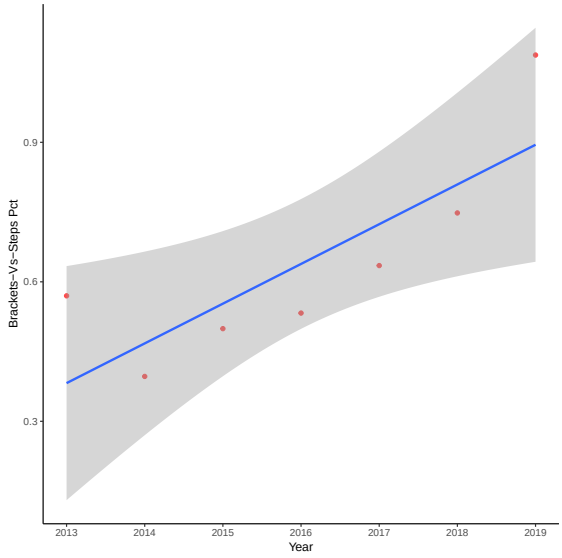
Table 5. Step cartographers who write charts densest in bracket-jumps.

Author name	#charts	BJJ%
Paul J Kim	45	71.8
Rust	97	51.7
bblum	40	49.3
Liam	11	44.5
Snooze	22	38.1
Rems	10	36.5
sssmsm	45	32.9
Loak	19	31.7
Little Matt	42	31.4
Paparazzi	10	29.6

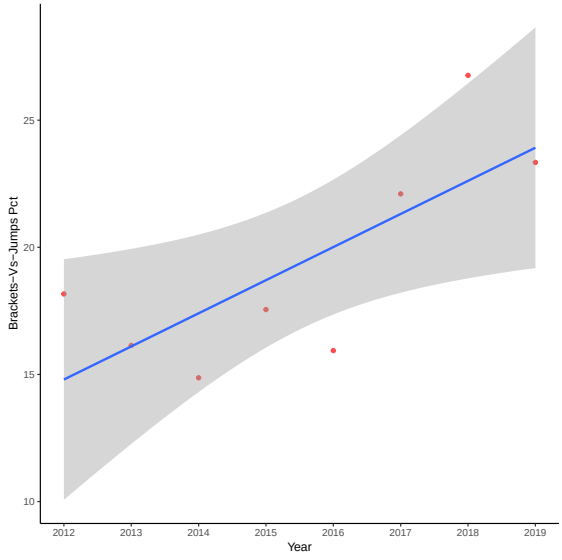
Table 6. Step cartographers whose jumps are most bracketable.

5. Discussion

I confirmed in stepmania the bracketiness of each of Section 4.1's high scorers, and happily observed no false positives (i.e., all charts seemed to intend the player to bracket), surprising myself for the 3rd time running at the accuracy of the algorithm. I did observe some false negatives, i.e., jumps seeming intended to be bracketed that the algorithm wouldn't catch. In Beach Party (Nero 2017), shown in Figure 7(a), the "mine jump" forces the player to reset her footing, removing the right foot's initial presence on U to allow LU to be bracketed. This would actually be pretty easy to fix (just have the algorithm parse mines like, at all) but it's the day of the deadline and I already ran the experiment, so. Another class of false positives showed up in koopa bling (Ali 2018) (Figure 7(b)), where



(a) BJS% across years ($\beta = 0.085\%$/yr, CI $= \{0.032, 0.14\}$).



(a) BJJ% across years ($\beta = 1.30\%$/yr, CI $= \{0.40, 2.21\}$).

Figure 6. Linear best fit of bracket jumps' popularity across the years of ITG's history. Grey area depicts 95% CI.

one foot is fixed to a hold, affecting the footing of subsequent jumps.[2] For kicks, I also show the source of the "I'm sorry" stepchart description in Figure 7(c), from Zombie Sunset (Sorry 2016) (Table 2); note here the DUR bracket-triple-jumps in the third measure, which the algorithm currently cannot process.

Many old charts written without regard to jump patterning allow for many of their jumps to be bracketed anyway. I noticed an old stepchart for One-Winged Angel (植松伸夫 1997) ranking high in the list of overall total bracket-jumps, with 182, simply because it's long and has a lot of steps, but upon playing, the patterns definitely do not feel intentional, and it even includes a section of stream with wholly unbracketable jumps interspersed. Figure 7(d) shows this section; in this stream, the player would expect to step the red and blue arrows with her right foot and the lime ones with her left, but note how the jumps (hint: all in blue arrows, unless you're reading this on dead trees) occur as a mix of left and right arrows, with even a UD "candle" jump making an appearance. However, it measures an unremarkable 9 BJS% and 38 BJJ%—I just wanted to feature it as an example of what non-bracketable jump stream looks like.

Authors seem to realize when their charts are too bracket jumpy for comfort: several of the charts appearing in the highest ranks of the spreadsheet had their step author field filled in as "Stupid" or "I'm sorry". In fact, I personally found the charts with 100 BJJ%, i.e., *all* jumps appearing therein were bracketable, to be more tasteful than those with more total BJS% but a few normal jumps as well. I'd hypothesize this is because to achieve 100 BJJ% requires a certain intentionality, resulting in better chart design overall (note that such charts necessarily include no LR or DU jumps whatsoever). Either that, or the more uncomfortable ones include certain extremely turny/candley/doublesteppy patterns around their brackets that causes the algorithm to count them as normal jumps instead. Anyway, Figure 7(e) shows the chart with most bracket-jumps (35) among ones with 100 BJJ%, which I tried out for myself and found quite enjoyable (ATB 2018). Amusingly, Encore (Reen 2017) features a Hard chart in addition to its Challenge, which former ranks second place to Electrical Paradise among 100 BJJ%s with 29 brackets. Upon inspection, I found this chart identical to the Challenge (upcoming in Figure 9(a)), only the crossover brackets (to be discussed later in Section 6.2) having been replaced by easier, normal ones.

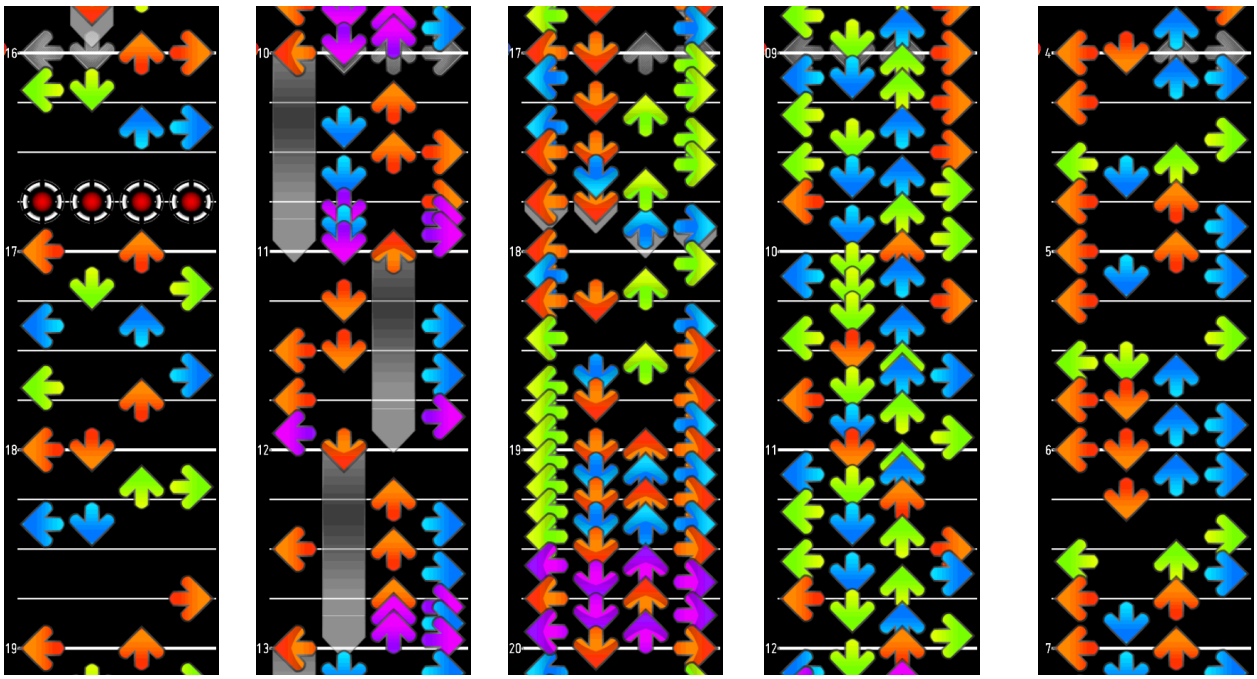
6. Never Work

The only thing that impresses the research community more than overdelivering on your future work promises is to overdeliver twice; hence, I shamelessly reuse the joke of this section name from last time.

6.1 UI

It would be cool to integrate this (and the preceding two algorithms (Blum 2016, 2017)) into the community's prevailing stepmania theme. Currently, as shown in Figure 8(a), the song preview screen wastes considerable space displaying the count of irrelevant chart aspects such as mines (now used primarily for signaling the pres-

[2] The algorithm actually foots them correctly in this chart by sheer luck (an even number of preceding doublesteps maintains footing parity), but these would be harder to handle in general.



(a) Beach Party, 15 (Nero 2017)

(b) koopa bling, 69 (Ali 2018)

(c) Zombie Sunset, 17 (Sorry 2016)

(d) 1-Winged Angel, 14 (植松伸夫 1997)

(e) Electrical Paradise, 12 (ATB 2018)

Figure 7. Example real-world charts with (a) false negative jumps that mines render bracketable after all, (b) jumps whose footing is affected by preceding holds fixing one foot in place, (c) brackets the chart author was sorry about, (d) "retro" style unbracketable jump-stream, and (e) intentionally many true bracket jumps from the BJJ% category winner.

(a) Current state-of-the-art.

(b) Artist's conception.

Figure 8. Song info preview panel for a recent bracket jump-heavy stepchart (Sanchez 2018).

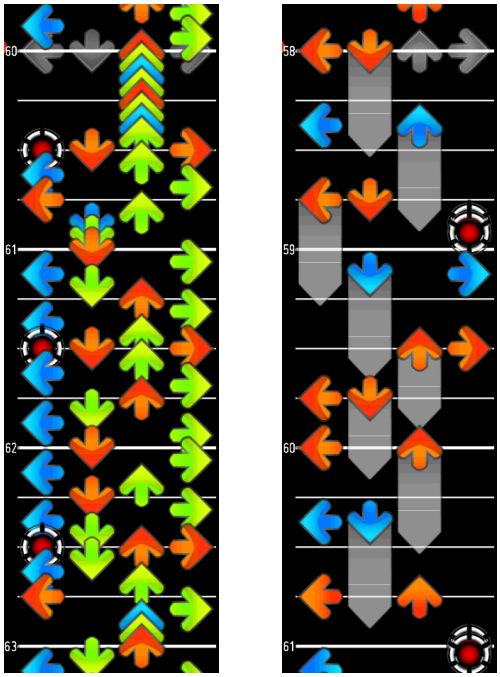
ence of footswitches) and hands (now mostly stepped with the feet by bracketing anyways). I bet the community might actually pay attention to my work—the holy grail of research, honestly—if popular themes used it to show precisely how technical a chart was, as in (b).

6.2 Crossover Brackets

Whereas with normal crossovers (e.g., in the sequence L-U-R, stepping the R with the left foot), the player's foot can pretty much point whichever way she finds most comfortable; however, crossover brackets (e.g., in the sequence L-U-DR...), the player's foot must point backwards (...stepping the DR's R with her left heel and D with toes), inducing extra turniness.[3] Such jumps could instead just be stepped with both feet as normal, possibly inducing doublesteps or jacks, and indeed that is how the algorithm presently handles them.

However, ambitious stepchart authors have recently experimented with encouraging the player to bracket-jumps while crossed-over. Such charts attempt to force, or at least hint, these jumps to be bracketed via additional stepchart elements: in Figure 9(a), mines on the left arrow force the player to remove her left foot in preparation for the crossover (and being in the middle of fast stream, further discourages doublestepping), and in Figure 9(b), extending one of the arrows as a hold encourages the player to bracket the subsequent jump with the other foot. In (b)'s case, the player must also switch feet on each of the non-hold arrows, effectively making these crossover-footswitch-brackets, which would truly be a wonder to identify programmatically. Note however the high measure counter in both, meaning these occur quite late in

[3] The other way, e.g. stepping the D with the left heel and R with toes, while your right foot is fixed on U, is extremely uncomfortable at speed. Trust me on this one.



(a) Encore, 12 (Reen 2017)

(a) Atariwave, 10 (Tuuc 2018)

Figure 9. Crossover bracket-jumps must be signaled via either (a) mines or (b) holds to remove ambiguity.

the song; both charts first spend some time "teaching" the player both to expect bracket jumps and how it intends to signal crossovers in general—so if a human is not expected to understand them at first glance, I think it is fair to leave out of the algorithm too for now.

6.3 Footswitch Brackets

I put this section here only just so I could have something to forward-reference back in Section 3, but not to actually write anything. I always wanted to do that, you know? I mean, footswitch brackets are a real thing, but still. ¯_(ツ)_/¯

6.4 Obligatory Machine Learning Section

I guess you could skip all this fiddly "algorithm" stuff by hopping on the pads yourself for a few rounds, playing a variety of technical charts, and just training a neural network based on how you stepped the patterns. It would probably even learn to fake crossovers more at the ends of songs than at the beginnings, where you're more likely to be physically tired. And like, do you really want that kind of bias in your dataset?

Remember kids, friends don't let friends use ML for problems that are more fun to solve by hand.

7. Conclusion

Bracket jumps is, in fact, statistically significantly, the new crossovers or something.

Acknowledgments

neuropantser (Michael T. Lawson) provided invaluable eleventh-hour (literally) advice on Section 4.2's statistical analysis, up to and including making the graphs for me. Greg Hanneman unknowingly helped copy-edit my hyphenation. And much love to the SIGBOVIK community.

References

m. Ali. koopa bling. UPS 4, 2018.

ATB. Electrical paradise. Chicago Timing Authority, 2018.

B. Blum. Which ITG stepcharts are turniest? SIGBOVIK, 2016.

B. Blum. A boring follow-up paper to "Which ITG stepcharts are turniest?" titled, "Which ITG stepcharts are crossoveriest and/or footswitchiest?". SIGBOVIK, 2017.

B. Blum. Practical concurrency testing, or: how I learned to stop worrying and love the exponential explosion. Ph.D. dissertation CMU-CS-18-128. Carnegie Mellon University, 2018.

w. dril. using my turn at a karaoke bar to try to do a 5 miunte routine that basically just says bryan singer is the new gawker writer or something. Twitter, 2019.

J. Nero. Beach party. Technical Showcase 2, 2017.

R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2018.

S. Reen. Encore. Technical Showcase, 2017.

S. Sanchez. Divine. Technical Showcase 3, 2018.

I. Sorry. Zombie sunset. Jimmy Jawns 2, 2016.

Tuuc. Atariwave. Technical Showcase 3, 2018.

H. Wickham. ggplot2: Elegant graphics for data analysis. Springer-Verlag New York, 2016. ISBN 978-3-319-24277-4.

植松伸夫. One-winged angel. Final Fantasy VII (stepchart pack and date unknown), 1997.