7-2016

Pythagorean Combinations for LEGO Robot Building.

Ronald I. Greenberg
Rgreen@luc.edu

Recommended Citation

Ronald I. Greenberg. Pythagorean combinations for LEGO robot building. In Proceedings of the Global Conference on Educational Robotics (GCER). KISS Institute for Practical Robotics, July 2016. http://files.kipr.org/gcer/2016/GCER_2016_Papers/Pythagorean_Combinations_fo r_Lego_Robot_Building.pdf

Ronald I. Greenberg[1] (rig@cs.luc.edu)
Pythagorean Combinations for LEGO Robot Building
Loyola University Chicago

Pythagorean Combinations for LEGO Robot Building

Abstract

This paper provides tips for LEGO® robot construction involving bracing or gear meshing along a diagonal using standard Botball® kits.

1 Introduction

Fred Martin gives many useful tips for LEGO robot construction, especially regarding use of gears [2]. But to mesh gears along a diagonal or construct diagonal bracing, he suggests experimentation, though he notes the obvious applicability of the Pythagorean Theorem. This paper provides a more systematic approach by using a spreadsheet to organize Pythagorean combinations that are close to exact and by noting some tricks that can be employed to place parts at spacings of half LEGO units (or sometimes even quarter units) using a standard Botball [1] kit (http://botballstore.org/product/botball-lego-bag). Martin also suggests some constructions involving fractional LEGO units, but these spacings are primarily achieved by using LEGO bricks, which are very scarce in the the Botball kit. The constructions in this paper instead use liftarms as the main structural pieces. (Specific liftarms and other pieces will be referenced below using names in the standard Botball parts list.)

It should be noted that the width and height of LEGO liftarms, as well as the space between adjacent holes along the length of a liftarm are all one LEGO unit. Meanwhile, the standard gears are of radius .5, 1, 1.5, and 2.5. In a typical alignment of gears along a liftarm, the gears of radius 1 will only mesh with each other (at a distance of 2), while the other gears may be abutted to one another in various combinations at distances of 1, 2, 3, 4, or 5. The most straightforward construction methodology is to place all liftarms horizontally or vertically, but the rest of this paper explores ways of achieving diagonal placement of liftarms and/or gears. (We actually have three dimensions in which to work, but at any given time, we focus on arranging components primarily within two of the dimensions.) Using diagonal liftarms may be a useful bracing mechanism that uses fewer liftarms, or, of potentially greater value, a mechanism to align gears (and transfer motion) along a diagonal. For diagonals of length at most 5, it may also be possible to place gear centers on the rectilinear grid but with the gears meeting along a diagonal.

2 Near-Integral Pythagorean Triples

We begin by exploring ways to construct diagonals with essentially no deviation from the underlying LEGO grid implied by the use of pieces with standard spacings.

[1] Supported in part by National Science Foundation grants CNS-1542971 and CNS-1543217.

An appropriately constructed spreadsheet provides a convenient way to view near-integral Pythagorean triples that may be helpful in LEGO construction. For example, the macro in Figure 1 initializes a Microsoft Excel® spreadsheet of relevant data, shown here with appropriate choices for integral lengths only ($FRAC = 1$ and $HYPRND = 1$).

Additional parameter choices in the code of Figure 1 were to limit the lengths of all triangle sides to 14 (longest available with a single liftarm) and to limit the slope (long leg divided by short leg) to 5, since constructions with slope closer to 1 are of greater interest here as opposed to essentially running along a horizontal or vertical. In addition to showing leg lengths, slope, and hypotenuse length, the spreadsheet shows the approximate hypotenuse length (rounded to the nearest multiple of $1/HYPRND$), the error relative to the actual hypotenuse length, and the absolute value of the error.

The macro of Figure 1 generates a spreadsheet with 68 data rows. Sorting by absolute error, we find 11 rows with absolute error of less than .1 LEGO units as shown in Figure 2. It is actually possible to construct nearly all of the 68 possibilities as verified by constructing cases near the bottom of the sorted spreadsheet, but the ones listed in Figure 2 will involve less deformation of the pieces. As an example, Figure 3(a) shows the case with least nonzero absolute error (7-11-13) used to hold a sequence of gears. (At least one combination remains unworkable, for example trying to build a 1-1-1 using a bent liftarm for the two legs.)

Probably, the most useful way to use the spreadsheet is to sort by slope after sorting by absolute error. Then when a particular slope is desired along which to align a sequence of gears, one may select the corresponding option with least absolute error. Figure 4 shows the triples with least absolute error for slopes from 1 to 5 at intervals of .5: As an example of this approach, the tightest construction of slope 1 is a 5-5-7. These dimensions can be used to create a tight construction while actually extending the hypotenuse farther to hold a longer sequence of gears; for example, see Figure 3(b).

The example constructions of Figure 3 use liftarms along the hypotenuse to hold gears securely in place, but one can also consider placing liftarms horizontally and vertically only while attempting to place two gear centers on such liftarms so that they mesh diagonally along the hypotenuse. In this case, the hypotenuse can be no longer than 5, the largest possible separation of adjacent gear centers (using gears of radius 2.5). The most obvious possibility is to use the exact Pythagorean triple 3-4-5. Even this exact triple may leave the gears able to slip past one another due to the possibility of axles wobbling in liftarm holes, etc., but with good bracing, it seems to be feasible also to work with the three triples of small hypotenuse that come next in absolute error: 1-5-5, 1-4-4, or even 1-3-3. Beyond this, the error switches sign and the gears bind, or the error is too great to ensure that the gears do not slip. Figure 5 shows the most extreme pair of cases from the four just mentioned.

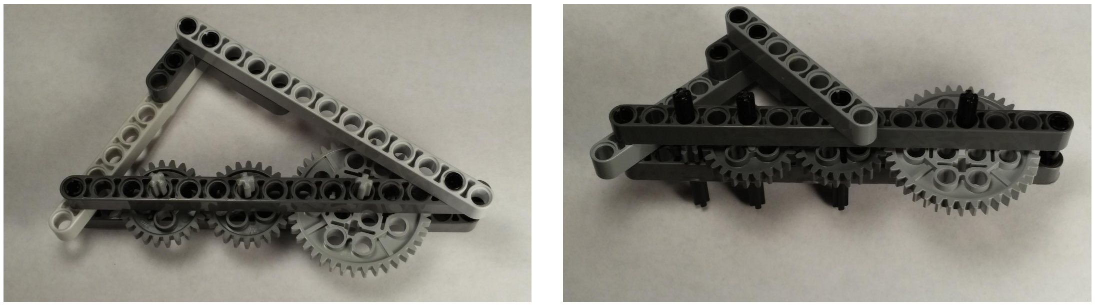
It may also be worthwhile to note that the “1×11.5 Liftarm Double Bent” pieces used in Figure 5(a) can be used to attach liftarms at a slope of 1. Additionally, the “1×9 Liftarm Bent (3×7)” and the “1×7 Liftarm Bent (4×4)” form an angle 90° greater than the small angle of a 3-4-5 triangle.

```
Sub Initialize()

Dim MAXLEG, MAXHYP, FRAC, HYPRND As Integer
MAXLEG = 14 ’Maximum allowed leg length
MAXHYP = 14 ’Maximum allowed hypotenuse length
MAXSLOPE = 5 ’Maximum allowed slope (long leg divided by short leg)
FRAC = 1 ’Fractions of Lego units allowed
         ’(1 for whole units only, 2 for halves, 4 for quarters, etc.)
HYPRND = 1 ’Approximate hypotenuse by rounding to specified fraction
           ’(1 for whole units, 2 for halves, 4 for quarters, etc.)

Cells.Clear
Cells(1, 1).Value = "Short Leg"  ’A1
Cells(1, 2).Value = "Long Leg"   ’B1
Cells(1, 3).Value = "Hypotenuse" ’C1
Cells(1, 4).Value = "Approx Hyp" ’D1
Cells(1, 5).Value = "Error"      ’E1
Cells(1, 6).Value = "Abs Err"    ’F1
Cells(1, 7).Value = "Slope"      ’G1

Dim i, j, Row As Integer
Row = 2
For i = 1 To MAXLEG * FRAC
   For j = i To MAXLEG * FRAC
      Cells(Row, 1).Value = i / FRAC ’A Row
      Cells(Row, 2).Value = j / FRAC ’B Row
      Cells(Row, 3).Formula = "=SQRT(A" & Row & " ^2 + B" & Row & "^2)"  ’C Row
      Cells(Row, 4).Formula = "=MROUND(C" & Row & "," & 1 / HYPRND & ")" ’D Row
      Cells(Row, 5).Formula = "=D" & Row & "-C" & Row                     ’E Row
      Cells(Row, 6).Formula = "=ABS(E" & Row & ")"                        ’F Row
      Cells(Row, 7).Formula = "=B" & Row & "/A" & Row                     ’G Row
      If Cells(Row, 4) <= MAXHYP And Cells(Row, 7) <= MAXSLOPE Then Row = Row + 1
   Next j
Next i
Rows(Row).EntireRow.Delete ’Remove last row not meeting condition at end of loop

End Sub
```

Figure 1: A Visual Basic for Applications macro to initialize a Microsoft Excel spreadsheet of Pythagorean triple data.

Short Leg	Long Leg	Hypotenuse	Approx Hyp	Error	Abs Err	Slope
3	4	5	5	0	0	1.333333333
5	12	13	13	0	0	2.4
6	8	10	10	0	0	1.333333333
7	11	13.03840481	13	-0.03840481	0.03840481	1.571428571
8	9	12.04159458	12	-0.041594579	0.041594579	1.125
4	8	8.94427191	9	0.05572809	0.05572809	2
4	7	8.062257748	8	-0.062257748	0.062257748	1.75
5	5	7.071067812	7	-0.071067812	0.071067812	1
5	13	13.92838828	14	0.071611723	0.071611723	2.6
5	11	12.08304597	12	-0.083045974	0.083045974	2.2
1	5	5.099019514	5	-0.099019514	0.099019514	5

Figure 2: The 11 near-integral (or integral) Pythagorean triples with sides ≤ 14 with least absolute error.



(a) (b)

Figure 3: The approximate Pythagorean triples with sides ≤ 14 with (a) least absolute error (7-11-13) and (b) least absolute error for a hypotenuse slope of 1 (5-5-7). The hypotenuse in (b) is extended to provide more space for gears.

Short Leg	Long Leg	Hypotenuse	Approx Hyp	Error	Abs Err	Slope
5	5	7.071067812	7	-0.071067812	0.071067812	1
6	9	10.81665383	11	0.183346174	0.183346174	1.5
4	8	8.94427191	9	0.05572809	0.05572809	2
4	10	10.77032961	11	0.229670386	0.229670386	2.5
1	3	3.16227766	3	-0.16227766	0.16227766	3
2	7	7.280109889	7	-0.280109889	0.280109889	3.5
1	4	4.123105626	4	-0.123105626	0.123105626	4
2	9	9.219544457	9	-0.219544457	0.219544457	4.5
1	5	5.099019514	5	-0.099019514	0.099019514	5

Figure 4: The approximate Pythagorean triples with sides ≤ 14 with least absolute error for slopes of 1 to 5 at intervals of .5.

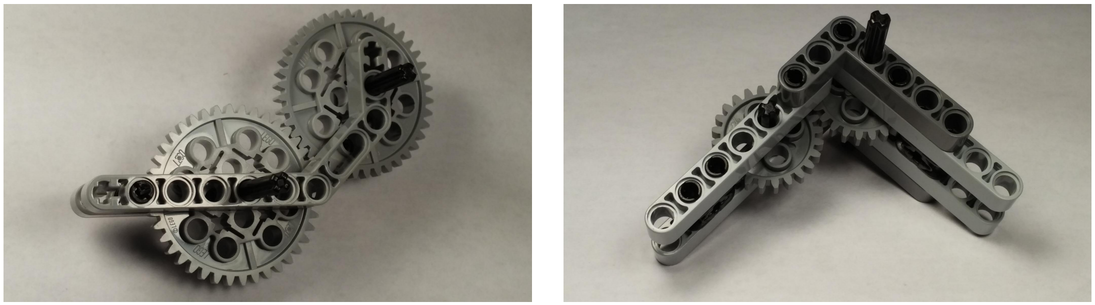

(a)
(b)

Figure 5: (a) The exact Pythagorean triple 3-4-5 with the hypotenuse formed solely by gears and (b) the approximate Pythagorean triple 1-3-3 that has the greatest absolute error that seems workable in practice.

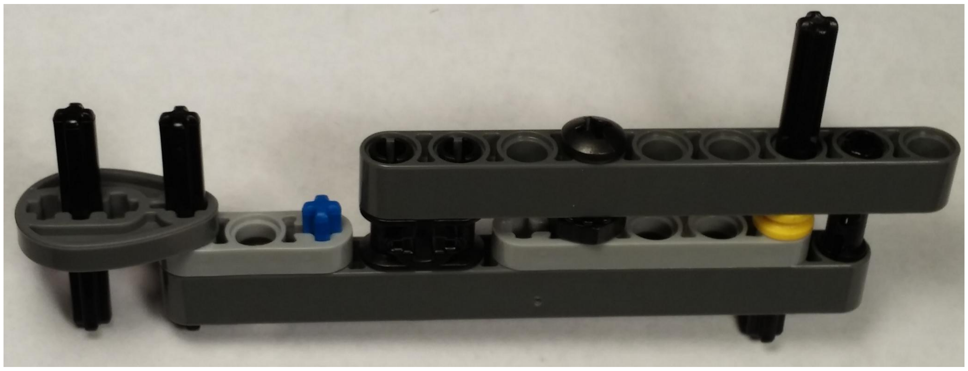


Figure 6: An illustration of various ways to incorporate half-unit spacing along the length of a liftarm or in a perpendicular direction.

3 Half-unit spacing

The left-hand side of Figure 6 shows a simple construction using a Cam and a "1×3 Liftarm Thin" that can be used to extend any ordinary liftarm by 1.5 units (to the leftmost axle center). Looking also at the right-hand side of Figure 6, we can see that cams, thin lift arms (including "Triangle pieces"), the "Bush 1/2" and "Nut 8-32 Keeps (black)" all can be used to create half-unit spacing perpendicular to a liftarm.

Tweaking our macro (Figure 1) for half-unit spacings up to 15.5 (longest liftarm extended by 1.5 units as in Figure 6) by setting $MAXLEG = 15.5$, $MAXHYP = 15.5$, $FRAC = 2$ and *HYPRND=2*, we obtain 301 data rows if we eliminate sides of length just .5. Extracting the triples with least absolute error for slopes from 1 to 5 at intervals of .5, we obtain Figure 7. All but the last of these is new in comparison to Figure 4.

As in Section 2, we can again consider using only gears (no liftarm) along the hypotenuse. We already know standard gear pairings achieve gear center spacings of 1, 2, 3, 4, or 5, so we can now consider an expanded set of triples with integral hypotenuse from 1 to 5. We also can use a new trick to place gears with center spacings in half-units. Specifically, the double bevel gears, though recommended by Martin [2] only for changing the angle of rotation of

Short Leg	Long Leg	Hypotenuse	Approx Hyp	Error	Abs Err	Slope
6	6	8.485281374	8.5	0.014718626	0.014718626	1
5	7.5	9.013878189	9	-0.013878189	0.013878189	1.5
2	4	4.472135955	4.5	0.027864045	0.027864045	2
5	12.5	13.46291202	13.5	0.037087982	0.037087982	2.5
3	9	9.486832981	9.5	0.013167019	0.013167019	3
4	14	14.56021978	14.5	-0.060219779	0.060219779	3.5
3.5	14	14.43086969	14.5	0.06913031	0.06913031	4
1	4.5	4.609772229	4.5	-0.109772229	0.109772229	4.5
1	5	5.099019514	5	-0.099019514	0.099019514	5

Figure 7: The approximate Pythagorean triples with sides ≤ 15.5 with least absolute error for slopes of 1 to 5 at intervals of .5 when half-unit side lengths are allowed.

shafts in a gear train, can mesh with each other like traditional gears do as long as we allow new gear center spacings. The three types of double bevel gears (12, 20, and 36 teeth), with radii of approximately .75, 1.25, and 2.25 units, can mesh with one another at distances of 1.5, 2, 2.5, 3, 3.5, or 4.5. Considering these and the traditional gear spacings as possible hypotenuse values yields 32 data rows. Figure 8 shows the 20 rows with the least absolute error, ending with the familiar 1-3-3 considered at the end of Section 2, beyond which the error seems too high to be prudent.

4 Quarter-unit spacing

Quarter-unit spacing is not generally easy to achieve, but we can consider using certain gear combinations to form a hypotenuse that measures in quarter units. Specifically, we can combine a traditional gear with a double bevel gear to obtain a spacing of 1.25, 1.75, 2.25, 2.75, 3.25, 3.75, or 4.75. Tweaking the macro of Figure 1 as in the previous section but with $HYPRND = 4$, we obtain the approximate Pythagorean triples of Figure 9 with a hypotenuse that can be formed by meshing a traditional gear with a double bevel gear.

5 Conclusion

The possible constructions indicated in this paper are by no means exhaustive. Various odd spacings may also be achieved by interposing metal pieces, bricks, or plates/tiles between liftarms, but the constructions here use the more plentiful pieces in a standard Botball kit and retain a regular grid-based approach, either on the traditional grid or on a half-unit grid.

References

[1] KISS Institute for Practical Robotics. Botball® educational robotics program. `http://www.botball.org`, 2015. Accessed 6/8/2016.

[2] Fred G. Martin. The art of LEGO design. *The Robotics Practitioner: The Journal for Robot Builders*, 1(2), Spring 1995.

Short Leg	Long Leg	Hypotenuse	Approx Hyp	Error	Abs Err	Slope
1.5	2	2.5	2.5	0	0	1.333333333
3	4	5	5	0	0	1.333333333
2	4	4.472135955	4.5	0.027864045	0.027864045	2
2	3.5	4.031128874	4	-0.031128874	0.031128874	1.75
2.5	2.5	3.535533906	3.5	-0.035533906	0.035533906	1
3.5	3.5	4.949747468	5	0.050252532	0.050252532	1
2	4.5	4.924428901	5	0.075571099	0.075571099	2.25
1.5	2.5	2.915475947	3	0.084524053	0.084524053	1.666666667
1	1	1.414213562	1.5	0.085786438	0.085786438	1
2.5	3	3.905124838	4	0.094875162	0.094875162	1.2
1	5	5.099019514	5	-0.099019514	0.099019514	5
2	3	3.605551275	3.5	-0.105551275	0.105551275	1.5
1	4.5	4.609772229	4.5	-0.109772229	0.109772229	4.5
3	3.5	4.609772229	4.5	-0.109772229	0.109772229	1.166666667
1.5	1.5	2.121320344	2	-0.121320344	0.121320344	1
1	4	4.123105626	4	-0.123105626	0.123105626	4
1	3.5	3.640054945	3.5	-0.140054945	0.140054945	3.5
1.5	3	3.354101966	3.5	0.145898034	0.145898034	2
2.5	4.5	5.14781507	5	-0.14781507	0.14781507	1.8
1	3	3.16227766	3	-0.16227766	0.16227766	3

Figure 8: The approximate Pythagorean triples with short hypotenuse with least absolute error while allowing half units.

Short Leg	Long Leg	Hypotenuse	Approx Hyp	Error	Abs Err	Slope
1.5	4.5	4.74341649	4.75	0.00658351	0.00658351	3
1	2	2.236067977	2.25	0.013932023	0.013932023	2
2.5	4	4.716990566	4.75	0.033009434	0.033009434	1.6
1	1.5	1.802775638	1.75	-0.052775638	0.052775638	1.5
1	2.5	2.692582404	2.75	0.057417596	0.057417596	2.5
1.5	3.5	3.807886553	3.75	-0.057886553	0.057886553	2.333333333
2	2	2.828427125	2.75	-0.078427125	0.078427125	1
1	3.5	3.640054945	3.75	0.109945055	0.109945055	3.5

Figure 9: The approximate Pythagorean triples with a hypotenuse that can be formed by meshing a traditional gear with a double bevel gear in order of least absolute error.