Solving Knapsack by OR Tools and Genetic Algorithm

Truong Minh Chau

May 2021

Contents

1 Knapsack problem:

The knapsack problem is a combinatorial optimization problem given a set of items, each with a weight and a value. The mission is to pack the volume of each item into a container so that the total weight is less than or equal a given capacity, whilst the total value is encouraged to include as large as possible. The problem often arises in resource allocation where the decision makers have to choose from a set of non-divisible projects or tasks under a fixed budget or time constraint, respectively.

2 Genetic Algorithm

Five stages are considered in this algorithm:
- Initial Population
- Fitness Calculation
- Selection
- Crossover
- Mutation

Modeling Knapsack problem with n items:
- Create the initial population of size n (generation 0)
- The fitness function in Knapsack problem is defined as how big value we get for a certain capacity
- Tournament selection with tournament size of 3
- The crossover is single-point crossover
- Flip-bit mutation with independent probability is $\frac{1}{n}$ for each attribute to be flipped

Then, setting up control parameters:
- Population size = length of Knapsack
- Probability for crossover = 0.9
- Probability for mutating an individual = 0.1
- Max generations = 50
- Ham of fame size = 1

3 Comparison

13 groups of Knapsack test used by OR tools and Genetic Algorithm to see how each method ran. Overall, the initial impression from the table below is that OR tools in most cases experienced better results in value or even if time, whereas the reverse is true for Genetic Algorithm but it does not mean this situation is always true because we still not tuned max generations. In addition, some group test OR tools can not return results as well as being over time allowed

	n	method	weight	value	time
00 - Uncorrelated	50	or	14721	20995	0.000953436
		deap	13487	18269	0.226156235
	100	or	252480	400811	0.005999804
		deap	22577	39444	0.727178335
	200	or	84317	50302	0
		deap	48785	76477	2.614793062
	500	or	118693	207992	0.000998259
		deap	130911	188344	12.07359958
	1000	or	243069	418472	0.004999399
		deap	242405	361459	43.7893219
01 - WeaklyCorrelated	50	or	14232	15768	0.001000166
		deap	11788	12910	0.267638206
	100	or	29013	31064	0
		deap	25369	27234	0.864213705
	200	or	51563	56976	0.000997543
		deap	48106	53157	2.393191338
	500	or	127276	139258	0.002997875
		deap	123758	135139	12.30732989
	1000	or	245972	273052	0.006998539
		deap	254758	270987	43.41770434
02 - StronglyCorrelated	50	or	14239	17539	0.018000603
		deap	11791	14991	0.274757624
	100	or	29017	35617	0.124008656
		deap	25388	31888	0.718029261
	200	or	51563	65363	301.0206373
		deap	48105	61305	2.22889924
	500	or	127278	162178	301.051019
		deap	123752	156852	11.82507062
	1000	or	245972	316372	301.1311274
		deap	254775	316875	42.59307432
03 - InverseStronglyCorrelated	50	or	16714	14914	2.601621866
		deap	14263	12463	0.274361372
	100	or	33968	30468	0
		deap	30341	26841	0.799145699
	200	or	61464	54964	0.015628099
		deap	58001	51101	2.16490674
	500	or	152031	136031	301.064687
		deap	148502	130802	11.30186462
	1000	or	295477	263977	301.0392482
		deap	304278	266078	39.97618461
04 - AlmostStronglyCorrelated	50	or	14238	17556	0.015627861
		deap	15171	15171	0.256643295
	100	or	29016	35611	0.041711807
		deap	31835	31835	0.984004974
	200	or	51563	65385	301.0110118
		deap	48093	61488	2.383163929
	500	or	127278	162154	301.0168614
		deap	156406	156406	12.35012174
	1000	or	245972	316415	0.235674143
		deap	317265	317265	45.45712709

05 - SubnetSum	50	or	14239	14239	0.001998663
		deap	11793	11793	0.270735979
	100	or	29017	29017	0.001001358
		deap	25391	25391	0.759972572
	200	or	51563	51563	0
		deap	48106	48106	2.113895178
	500	or	127278	127278	0.002003193
		deap	123764	123764	12.30979252
	1000	or	245972	245972	0.003995895
		deap	254777	254777	48.19847989
06 - UncorrelatedWithSimilarWeights	50	or	2401482	19676	0.047006607
		deap	2401292	17312	0.313180447
	100	or	4902253	39791	8.275791645
		deap	4902202	36480	0.98621726
	200	or	9904900	75678	0
		deap	9905130	70220	2.647738934
	500	or	24712055	189769	301.0568092
		deap	24713094	174147	13.57679105
	1000	or	49525319	371246	0.062517881
		deap	49523479	336219	42.27430511
07 - SpannerUncorrelated	50	or	4569	13472	0
		deap	6578	11948	0.254042625
	100	or	N/A	N/A	N/A
		deap	11914	22147	1.12498951
	200	or	N/A	N/A	N/A
		deap	25898	43340	2.292871237
	500	or	N/A	N/A	N/A
		deap	63365	113366	11.71095991
	1000	or	N/A	N/A	N/A
		deap	120497	210026	41.7490077
08 - SpannerWeaklyCorrelated	50	or	11452	10354	1.598035574
		deap	6502	15236	0.27847743
	100	or	20824	20550	0
		deap	11997	26796	0.823205948
	200	or	41116	40575	15.29231739
		deap	23703	59094	2.393257618
	500	or	N/A	N/A	N/A
		deap	64475	142890	13.36739898
	1000	or	N/A	N/A	N/A
		deap	125399	257802	40.96114373
09 - SpannerStronglyCorrelated	50	or	11540	28440	300.0217519
		deap	6461	27261	0.263941765
	100	or	20956	51656	11.11649275
		deap	12004	49204	0.811476946
	200	or	41288	101888	300.0227916
		deap	23674	102874	2.515035629
	500	or	99944	245144	302.3108802
		deap	64580	256080	11.43614721
	1000	or	198780	488680	300.0667527
		deap	125424	481624	43.22834778

10 - MultipleStronglyCorrelated	50	or	14238	21338	0
		deap	11764	19164	0.352093697
	100	or	29016	43316	1.799211264
		deap	25379	39879	0.783131838
	200	or	51558	81658	24.95177031
		deap	48098	77598	2.556943417
	500	or	127278	203778	300.1431954
		deap	123715	195915	11.4496882
	1000	or	245970	399170	300.0358379
		deap	254745	392945	41.33583736
11 - ProfitCeiling	50	or	14238	14229	1.476347208
		deap	11793	11769	0.459925652
	100	or	29015	29001	153.8344586
		deap	25389	25356	0.718583345
	200	or	N/A	N/A	N/A
		deap	48106	48048	2.23585701
	500	or	N/A	N/A	N/A
		deap	123764	123615	10.93139982
	1000	or	N/A	N/A	N/A
		deap	254776	254490	39.52536559
12 - Circle	50	or	14239	300031	0.651958704
		deap	11793	248484	0.280065298
	100	or	29017	611418	0.183720112
		deap	25391	535000	0.773991346
	200	or	N/A	N/A	N/A
		deap	48106	1013616	2.324188232
	500	or	N/A	N/A	N/A
		deap	123764	2607767	11.04529357
	1000	or	N/A	N/A	N/A
		deap	254777	5368272	40.01817894

4 Genetic Algorithm after tuning parameters

After increasing Max Generations to 200, I got a better overall result but with a small n, it shared comparable figures. Albeit negligible good as OR tools, increasing max-generation and max-population large enough, the results I got can be better. In addition, we can try to tune another parameters to find an optimal pattern.

	n	weight	value	time	maxgen
12 - Circle	50	11793	248484	0.28	50
		11792	248461	1.13	200
	100	25391	535000	0.77	50
		25391	535001	2.53	200
	200	48106	1013616	2.32	50
		48106	1013617	7.32	200
	500	123764	2607767	11.05	50
		123764	2607790	40.57	200
	1000	254777	5368272	40.02	50
		254777	5368323	158.25	200

5 Summary

In conclusion, these data sources clearly communicate that the result is not exactly optimal, it just close to the optimal one. So I believe that if I let OR tools and Genetic Algorithm run long enough or with optimal parameters, it may return the optimal result.