

トーナメント選択におけるトーナメント選出個体数に関する考察

Number of Participants in Tournament Selection

中田 和樹¹ 片田 喜章^{1*}
Kazuki NAKATA¹ Yoshiaki KATADA¹

¹ 摂南大学
¹ Setsunan University

Abstract: Tournament selection has been employed in various evolutionary algorithms due to its own advantages. Such a characteristic as to choose some individuals randomly in a current population for tournaments would make it possible to evaluate the only individuals chosen for those tournaments. This would be a new advantage of tournament selection in shortening the evaluation time. This study investigated this possibility analytically and experimentally. The results show that it is possible to shorten the computational time by evaluating only individuals chosen for tournaments if the tournament size is less than ten.

1 はじめに

トーナメント選択は個体の適応度の上下関係だけを用いるため、スケーリングの必要がなく、また、トーナメントサイズを変えることで選択圧を調整することができるという特徴をもつ [1][2]。トーナメント選択の実行には、トーナメントに選出される親個体候補だけの適応度の情報が必要であり、必ずしもすべての個体の適応度の情報を必要としない。これは、トーナメント選出個体のみ適応度の計算を行うことで、個体群の評価にかかる計算コストを削減することができる可能性を示している。しかもこれに、パフォーマンスとのトレードオフは存在しない。このことは、個体の評価に非常に時間がかかる類いの実問題では、実際的な評価回数が減ることにつながり、実用面において非常に望ましいことである。トーナメント選出個体のみ適応度の計算を行えば、計算コストを削減することができるということは直感的にはわかるが、どれくらい削減できるかという定量的な解析は行われていないように思われる。そこで本研究では、トーナメント選出個体数の期待値を、個体群の個体数およびトーナメントサイズをそれぞれ変更した場合の理論値と実験値を計算し、その比較を行う。また、トーナメント選択を用いた進化計算の実行過程における選出個体数の変動をテスト問題において調べる。

2 トーナメント選出個体数

2.1 トーナメント選択

本研究で取り扱う個体評価と選択に関して議論を始める前に、まず、GA を例として進化計算の基本的な手順を示す。

ステップ 1 初期個体群の生成

ステップ 2 個体群に含まれる個体の評価

ステップ 3 親個体の選択

ステップ 4 交叉・突然変異の適用

ステップ 5 終了条件が満たされていない場合は、ステップ 2 へ戻る。

トーナメント選択では、事前に設定されたトーナメントサイズ ($t \geq 2$) の個体が個体群からランダムにトーナメントに選出される。トーナメントに選出された個体のうち最良の適応度をもつ個体のみが次世代の親として選択される。個体数を一定に保つために、個体群の個体数 (N) が満たされるまでトーナメントを実施する。したがって、 N 回のトーナメントが実施され、1 世代で合計 (Nt) 個の個体が親個体候補としてトーナメントに選出される。トーナメントへの個体の選出には復元抽出と非復元抽出を行う場合とがあるが、いずれにしても、いくつかの個体は複数回選出される可能性がある。よって、上述のステップ 2 とステップ 3 を入れ替えることで選出された個体のみ評価すれば良く、すべての個体を評価する必要がなくなり、計算コストが削減できるというのが本研究の基本的なアイデアである。

*連絡先: 摂南大学理工学部電気電子工学科
〒 572-8508 大阪府寝屋川市池田中町 17-8
E-mail: katada @ ele.setsunan.ac.jp

2.2 選出個体数の導出

以下では、トーナメントへの個体の選出は復元抽出を行い、個体群に含まれる個体はすべて異なるものと仮定する。この場合、各個体は等しく $1/N$ の確率でトーナメントに選出される。よって、1 世代で実行される全トーナメントには、確率 $1/N$ で独立に (Nt) 個の個体が親個体候補として選出されることになる。これは、確率論でいう独立試行である。以上より、トーナメントへの選出個体が X 種類 (ただし、 $X \in \{1, 2, \dots, N\}$) である確率 $(P(X))$ は以下のように $P(X = 1)$ より順次計算できる¹。

$$\begin{aligned} P(X = 1) &= N \underbrace{\left(\frac{1}{N}\right)\left(\frac{1}{N}\right)\cdots\left(\frac{1}{N}\right)}_{Nt} \\ &= N \left(\frac{1}{N}\right)^{Nt} \end{aligned} \quad (1)$$

$$\begin{aligned} P(X = 2) &= \binom{N}{2} (2^{Nt} - 2) \left(\frac{1}{N}\right)^{Nt} \\ &\equiv \binom{N}{2} W_2 \left(\frac{1}{N}\right)^{Nt} \end{aligned} \quad (2)$$

$$\begin{aligned} P(X = 3) &= \binom{N}{3} \{3^{Nt} - \binom{3}{2} W_2 - 3\} \left(\frac{1}{N}\right)^{Nt} \\ &\equiv \binom{N}{3} W_3 \left(\frac{1}{N}\right)^{Nt} \end{aligned} \quad (3)$$

$$\begin{aligned} P(X = 4) &= \binom{N}{4} \left[4^{Nt} - \binom{4}{3} W_3 - \binom{4}{2} W_2 - 4 \right] \left(\frac{1}{N}\right)^{Nt} \\ &\equiv \binom{N}{4} W_4 \left(\frac{1}{N}\right)^{Nt} \end{aligned} \quad (4)$$

⋮

$$\begin{aligned} P(X = j) &= \binom{N}{j} \left[j^{Nt} - \binom{j}{j-1} W_{j-1} - \binom{j}{j-2} W_{j-2} - \cdots - \binom{j}{3} W_3 - \binom{j}{2} W_2 - j \right] \left(\frac{1}{N}\right)^{Nt} \\ &\equiv \binom{N}{j} W_j \left(\frac{1}{N}\right)^{Nt} \end{aligned} \quad (5)$$

⋮

$$\begin{aligned} P(X = N) &= \binom{N}{N} \left[N^{Nt} - \binom{N}{N-1} W_{N-1} - \binom{N}{N-2} W_{N-2} - \cdots - \binom{N}{3} W_3 - \binom{N}{2} W_2 - N \right] \left(\frac{1}{N}\right)^{Nt} \end{aligned} \quad (6)$$

式 (1)-(6) より $\sum_{j=1}^N P(X = j) = 1$ である。以上より、1 世代で全トーナメントに選出される個体数 (個体の種類) の期待値は以下で求められる。

$$\bar{X}_{theo} = E(X) = \sum_{j=1}^N j P(X = j) \quad (7)$$

つまり、式 (7) の期待値が個体群の個体数 (N) よりも小さいとき、上述の手順の入れ替えに効果があり、計算コストの削減が期待できる。

3 数値実験

本節では、第 2 節で導出したトーナメント選出個体数の確率分布から期待値を計算し、シミュレーションによって得られる期待値と比較する。

まず、シミュレーションの設定について述べる。2.2 節で述べた仮定に基づき異なる個体を N 個作成し、ランダムに Nt 個選出する。このとき、選出された個体の種類を X_i とする。これを n 回独立に試行して得られる結果 $\{X_1, X_2, \dots, X_n\}$ の平均 $\bar{X}_{sim} = \sum_{i=1}^n X_i / n$ を計算する。

$N \in \{5, 10, 25, 50, 100, 200, 500, 1000\}$, $t \in \{2, 3, \dots, N\}$, $n = 1000$ として計算を行った。ただし、式 (7) の計算では、大きい値をとる (N, t) の各組み合わせに対し式 (1)-(6) 中の値が非常に大きくなることから数値計算を実行することが難しい。本稿では Excel で計算できる範囲の結果を示す。

図 1, 表 1 に N を変化させたときの t に対する選出個体数 (\bar{X}_{theo} : 理論期待値, \bar{X}_{sim} : シミュレーション期待値) を示す。得られた範囲の理論値とシミュレーションの値は非常によく一致していることがわかる。第 2 節で述べたように、選出個体数の期待値 (以下、 \bar{X} と示す) が個体群の個体数 (N) よりも小さいとき、計算コストの削減が期待できる。 t が最小値 2 から増加していくと \bar{X} も増加し、 t が 10 以上で \bar{X} は N に収束している。 $N = 5, 10$ では N の値が 10 以下であるため、 t が N の値に近づくまで \bar{X} は増加する (図 1(a)-1(b))。 \bar{X} が N に収束する厳密な t の値は定かではないが、おおむね $t = 8$ 付近で収束している。つまり、第 2 節で述べた手順の入れ替えに効果があるのはトーナメントサイズが 1 桁の場合であり、現実的には $t \leq 4$ の場合であろう。

¹ただし、 $\binom{n}{k}$ は組合せの個数 ${}_n C_k$ を表す。

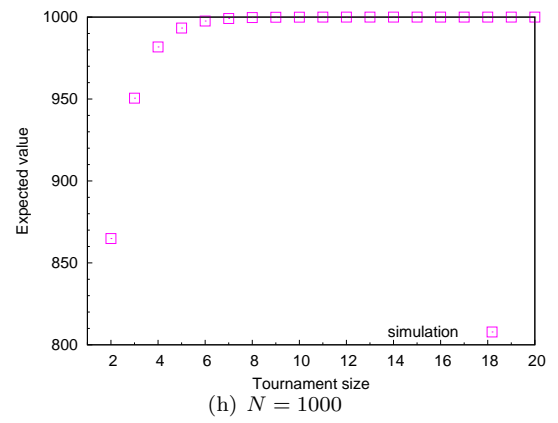
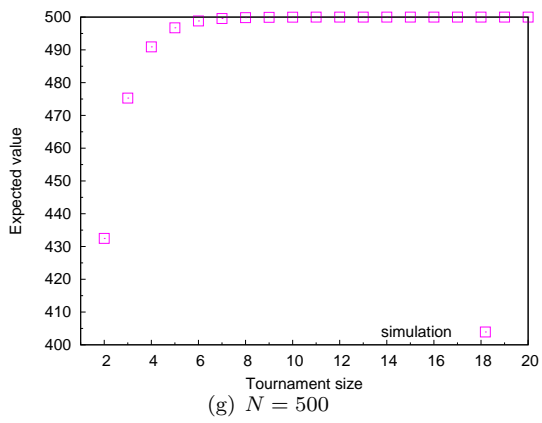
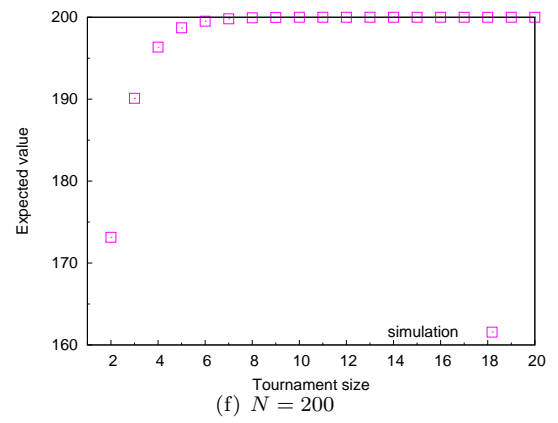
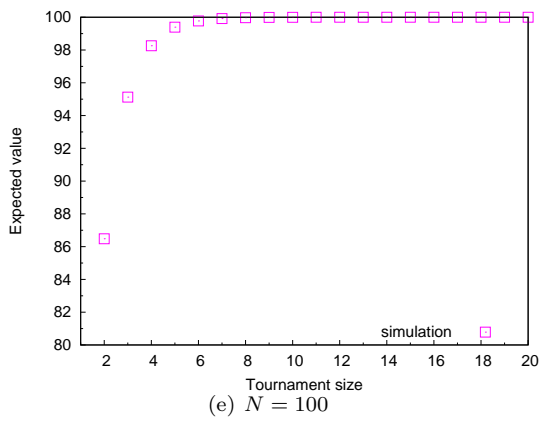
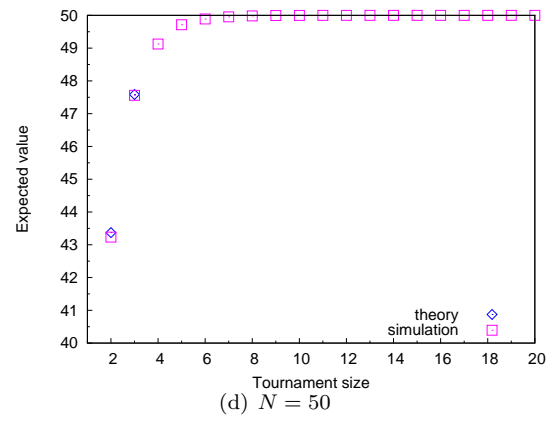
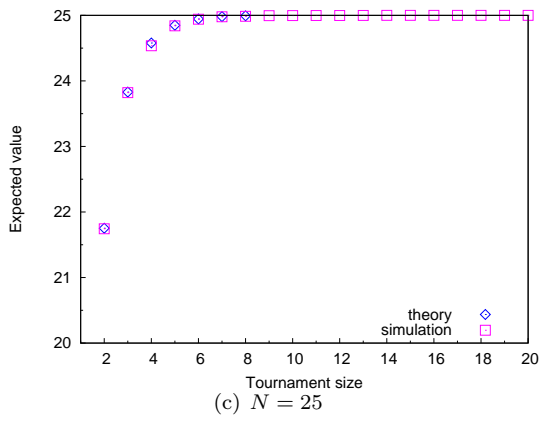
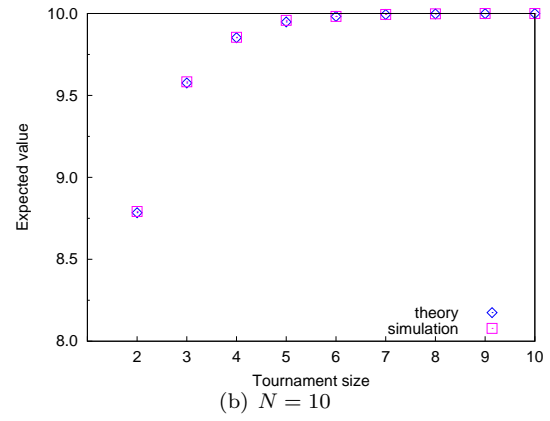
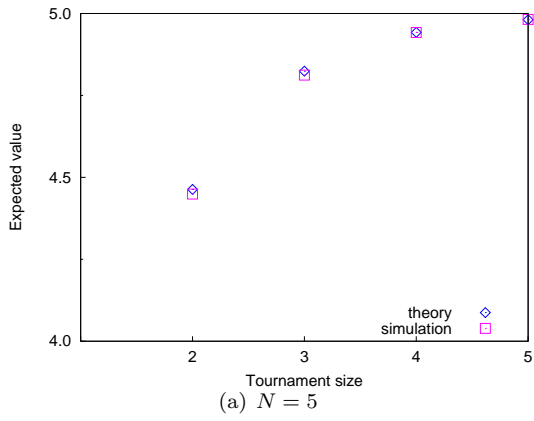


図 1: トーナメントサイズに対する選出個体数の期待値 (theory: 理論期待値, simulation: シミュレーション期待値)

表 1: (N, t) の各組合わせにおける選出個体数の期待値 (上段: \bar{X}_{sim} , 下段: \bar{X}_{theo})

t	$N = 5$	$N = 10$	$N = 25$	$N = 50$
2	4.4490	8.7920	21.7440	43.2340
	4.4631	8.7842	21.7529	43.3690
3	4.8120	9.5830	23.8210	47.5600
	4.8241	9.5761	23.8297	47.5852
4	4.9420	9.8550	24.5360	49.1260
	4.9424	9.8522	24.5782	—
5	4.9820	9.9580	24.8400	49.7130
	4.9811	9.9485	24.8480	—
6	—	9.9820	24.9410	49.8890
	—	9.9820	24.9452	—
7	—	9.9940	24.9780	49.9550
	—	9.9937	24.9803	—
8	—	9.9980	24.9850	49.9840
	—	9.9978	24.9929	—

N の値を変えて実験を行っているため、 \bar{X} は様々な値をとるが、 N からの削減率は t に対して同じような値をとる。これを示すために以下の削減率 RED [%] を定義する。

$$RED = \left(1 - \frac{\bar{X}}{N}\right) \times 100 \quad (8)$$

理論値 \bar{X}_{theo} は (N, t) のすべての組合わせに対しては値を計算できていないため、 RED の計算にはシミュレーションで得られた値 \bar{X}_{sim} を用いる。図 2, 表 2 に $t = 1, 2, \dots, 10$ の範囲の RED を示す。 $N = 5, 10$ で少しバラつきがあるものの、 RED は N の値に依らず t に対して近い値をとる。

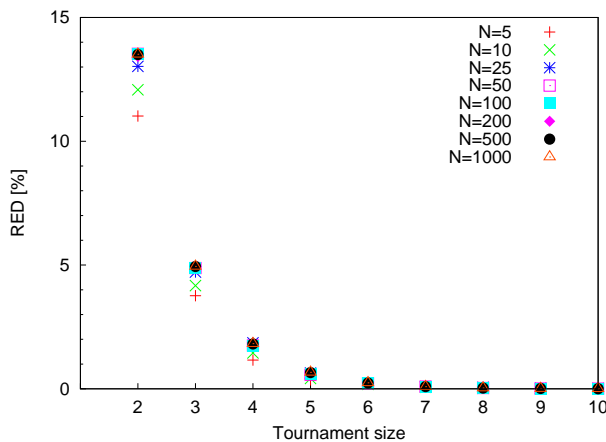


図 2: (N, t) の各組合わせにおける削減率 [%]

4 進化計算実験

本節では、トーナメント選択を用いた進化計算実行時の選出個体数を調べ、第 2 節および第 3 節の結果と比較する。進化計算にトーナメント選択を用いた単純 GA (以下、GA とよぶ) を採用し、テスト関数には ONEMAX 関数と (k, l) -DECEPTION 関数 [3] を用いる。ONEMAX 関数はバイナリーコーディングを用いた GA のテスト関数として最も簡単で単峰性を示し、 (k, l) -DECEPTION 関数は局所解をもつため、これらが個体群の構成、つまり、選出個体数に影響を与えるかを調べるために採用した。ONEMAX 関数はストリング長を $L = 100$ 、DECEPTION 関数は文献 [4] に従い、 $k = 3, l = 20$ の $L = 60$ とする。また、 (k, l) -DECEPTION 関数の設定を表 3 に示す。GA の各パラメータを表 4 のように設定し、個体数・トーナメントサイズを変更して、世代数に対するトーナメント選出個体数を調べた。これは、第 2.2 節で設定した、個体群に含まれる個体はすべて異なるという仮定が 1 世代目では成り立つが、それ以降の世代では GA のダイナミクスに依存して必ずしも成り立たず、その挙動を調べるためである。

GA を 100 試行行ったときの各世代に対する選出個体数の平均・最小値・最大値を図 3, 4 に示す。 (N, t) の各組合わせに対し、平均値は第 2 節および第 3 節で求めた値に世代を通してほぼ等しいことがわかる。選出個体数のゆらぎがどの程度か調べるために最小値・最大値を示したが、そのゆらぎの幅はトーナメントサイズ・個体数に依存していることがわかる。ただし、 (N, t) の各組合わせに対し 2 つの問題間ではそれほど差は見られない。 (k, l) -DECEPTION 関数の局所解の存在が個体群の収束をもたらすことで異なる個体数が減少し、選出個体数が減少するのではないかと予想したが、結果はそれに反するものであった。これは設定したトーナメントサイズの範囲が小さいもので収束をもたらすほどの選択圧ではなかったか、用いた 1 点交叉によってある程度の異なる個体数が確保されたためではないかと推測する。以上の結果より、GA 実行時においても、第 2 節および第 3 節で得たトーナメント選択を用いた場合の評価回数の削減率は評価回数 (個体数 \times 世代数)

表 3: $(3, l)$ -DECEPTION 関数

ストリング	適応度	ストリング	適応度
111	30	100	14
101	0	010	22
110	0	001	26
011	0	000	28

表 2: (N, t) の各組合わせにおける削減率 [%]

t	$N = 5$	$N = 10$	$N = 25$	$N = 50$	$N = 100$	$N = 200$	$N = 500$	$N = 1000$
2	11.0200	12.0800	13.0240	13.5320	13.5210	13.4260	13.5036	13.5159
3	3.7600	4.1700	4.7160	4.8800	4.8730	4.9450	4.9414	4.9480
4	1.1600	1.4500	1.8560	1.7480	1.7380	1.8205	1.8154	1.8238
5	0.3600	0.4200	0.6400	0.5740	0.6080	0.6455	0.6548	0.6668
6	—	0.1800	0.2360	0.2220	0.2270	0.2410	0.2344	0.2425
7	—	0.0600	0.0880	0.0900	0.0800	0.0960	0.0884	0.0868
8	—	0.0200	0.0600	0.0320	0.0300	0.0320	0.0342	0.0310

表 4: GA のパラメータ設定

世代数	100
点突然変異確率	$1/L$
1 点交叉確率	0.9
個体数 (N)	{100, 1000}
トーナメントサイズ (t)	{2, 4, 6}

に対して期待値として得られることが確認できた。

5 むすび

本稿では、トーナメント選出個体数の期待値を、個体群の個体数およびトーナメントサイズをそれぞれ変更した場合の理論値と実験値を求めた。以下に結果をまとめる。

- トーナメント選出個体数の確率分布を導出し、期待値を求めた。ただし、数値計算の過程で非常に大きな数値を扱うため、設定した個体群の個体数およびトーナメントサイズのすべての組合わせに対し理論式から値を計算できたわけではない。得られた範囲の理論値とその検証用計算機実験で得られた値は非常によく一致する。
- トーナメント選出個体数はトーナメントサイズが $t = 2$ のときに最小値をとり、 t が増加していくと増加していく。おおむね $t = 8$ 付近で個体群の個体数の値に収束する。また、個体群の個体数によらず、選出個体の削減率はトーナメントサイズに対し近い値をとる。
- 進化計算実行時の世代数に対する選出個体数は、理論式から得た値およびその検証用計算機実験で得られた値とよく一致する。

以上より、トーナメント選択による評価回数（より正確には評価個体数）の削減は、トーナメントサイズが小さいときに可能であることがわかった。したがって、選択圧が弱いときにパフォーマンスが良い進化計算には有効な方法であるが、選択圧が高い必要のある進化計算には向かない。

トーナメント選択を採用しない進化計算も多くある。これらに対しても、トーナメント選択を採用して同等のパフォーマンスが得られるならば、付加価値として評価回数の削減が期待できる。

本研究ではトーナメントへの個体の選出は復元抽出を行いた。非復元抽出を行いた場合、選出個体数（の種類）は最低でもトーナメントサイズ以上になり、その期待値は復元抽出の場合よりも高くなると予想される。また、進化計算実行時の実験に GA を用いた。一般的に多くの個体数を設定する進化計算（例えば GP）でも本稿で考察した内容は有効であると期待される。今後これらについて検証していく予定である。

参考文献

- [1] Goldberg, D.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley (1989)
- [2] 電気学会 進化技術応用調査専門委員会 (編): 進化技術ハンドブック 基礎編, 近代科学社, (2010)
- [3] Goldberg, D., Deb, K. and Korb, B.: Messy Genetic Algorithms Revisited: Studies in Mixed Size and Scale, *Complex Systems*, 4:415-444 (1990)
- [4] Mühlenbein, H.: Evolutionary Algorithms: Theory and Applications, *citeseer.nj.nec.com/110687.html*, (1993)

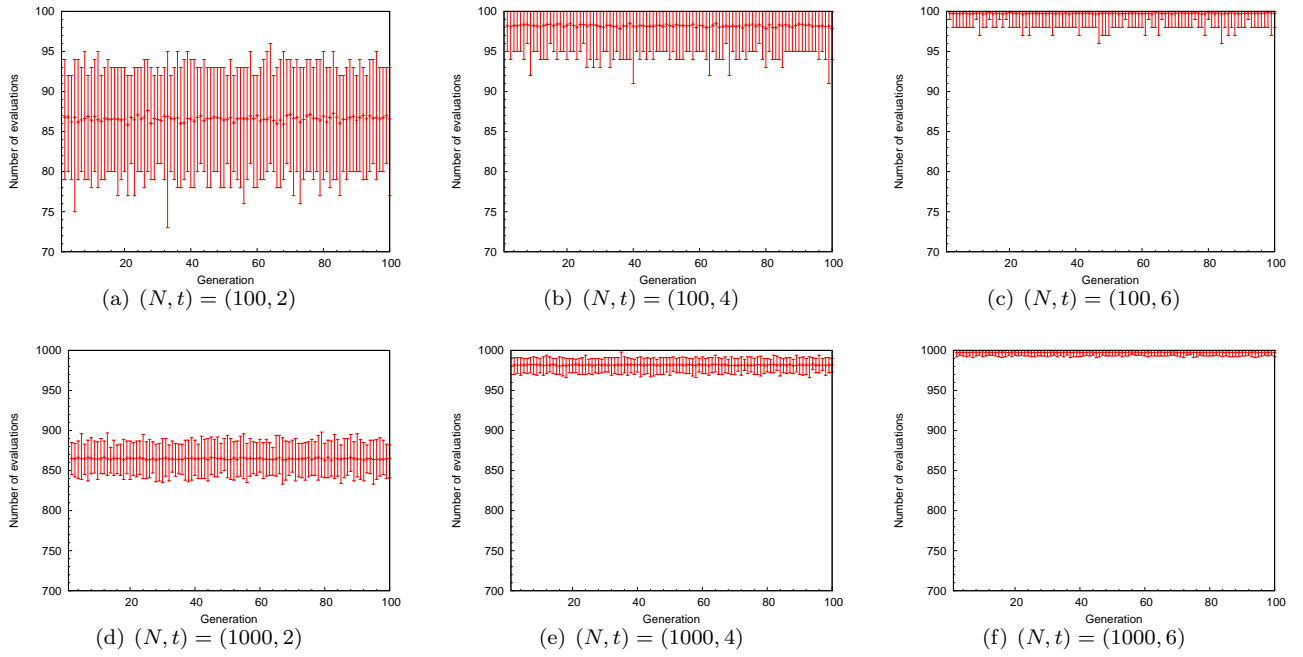


図 3: ONEMAX 関数における世代数に対する選出個体数 (最大値・平均値・最小値)

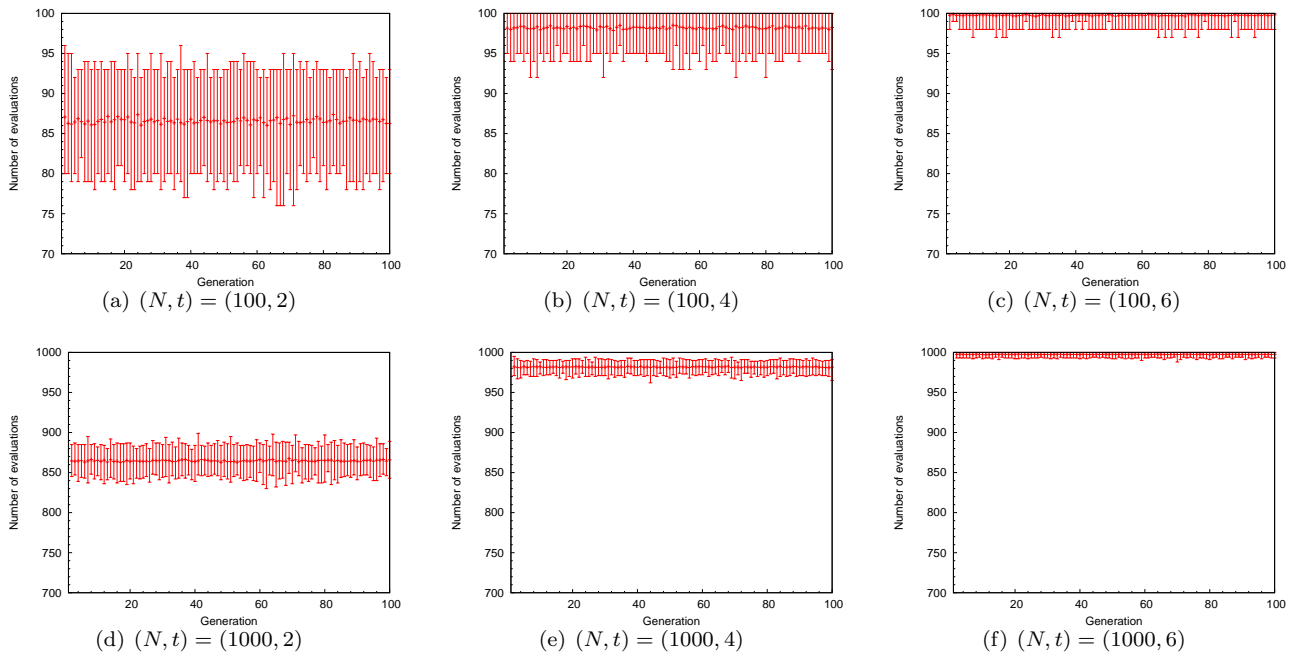


図 4: (3-20)-DECEPT 関数における世代数に対する選出個体数 (最大値・平均値・最小値)