Chapter 6
Fine Local Tuning

- Positional global knowledge:
  - Bits far to the right of such a sequence have quite a smaller influence
  - Bits far to the left of such a sequence have quite a significant

Mutation

- **Uniform mutation**

$$v_k^{'} \in [l_k, u_k]$$

- **Non-uniform mutation**: (fine tuning capabilities)

$$v_k = \begin{cases} v_k + \Delta(t, u_k - v_k) & \text{if a random digit is 0} \\ v_k - \Delta(t, v_k - l_k) & \text{if a random digit is 1} \end{cases}$$

$$\Delta(t, y) = y(1 - r^{(1-\frac{t}{T})^b})$$

where T: maximal generation number

b: a system parameter

$r \in [0..1]$

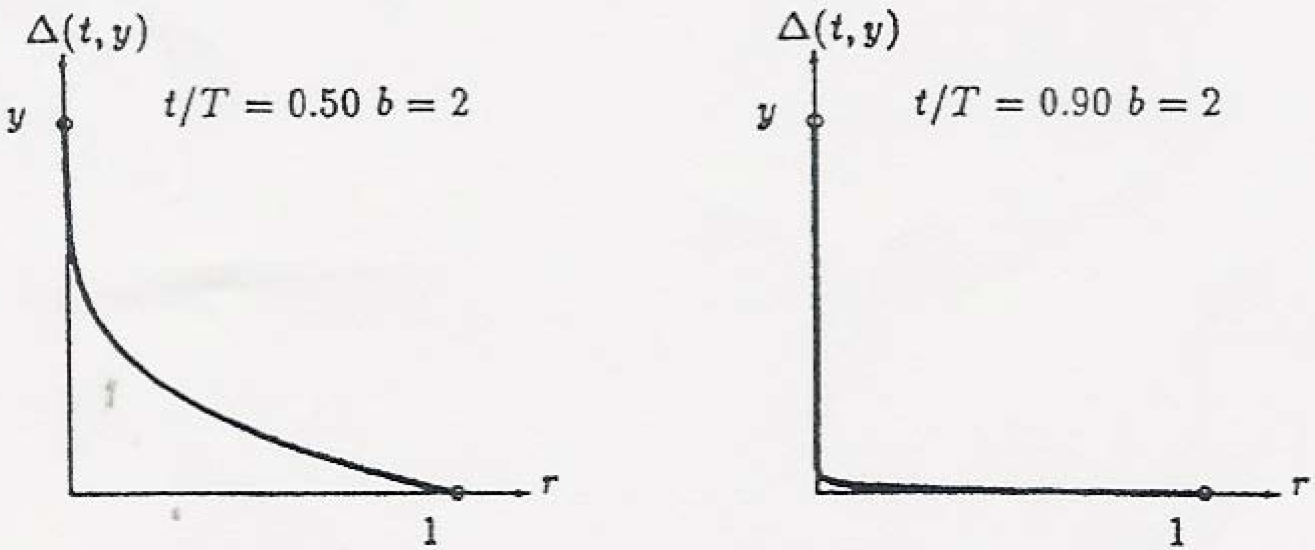

Fig. 6.1. $\Delta(t,y)$ for two selected times

Note: This property cause this operator to search the space uniformly initially (when *t* is small), and very locally at later stages.

- **Uniform vs. Non-uniform mutation**

Case	Exact solution value	GA w/ non-uniform mutation value	GA w/ non-uniform mutation D	GA w/o non-uniform mutation value	GA w/o non-uniform mutation D
I	16180.3399	16180.3939	0.000%	16234.3233	0.334%
II	109160.7978	109163.0278	0.000%	113807.2444	4.257%
III	10009990.0200	10010391.3989	0.004%	10128951.4515	1.188%
IV	37015.6212	37016.0806	0.001%	37035.5652	0.054%
V	287569.3725	287569.7389	0.000%	298214.4587	3.702%
VI	16180.3399	16180.6166	0.002%	16238.2135	0.358%
VII	16180.3399	16188.2394	0.048%	17278.8502	6.786%
VIII	10000.5000	10000.5000	0.000%	10000.5000	0.000%
IX	431004.0987	431004.4092	0.000%	431610.9771	0.141%
X	10000.9999	10001.0045	0.000%	10439.2695	4.380%

Table 6.9. Comparison of solutions for the linear-quadratic dynamic control problem

EP vs. Other Methods

- GA vs. GAMS (General Algebraic Modeling System)
  - Performance
  - Time
- Uniform vs. Non-uniform Mutation

GA vs. GAMS (General Algebraic Modeling System)

- Performance

Case	Exact solution value	Evolution Program value	Evolution Program D	GAMS value	GAMS D
I	16180.3399	16180.3928	0.000%	16180.3399	0.000%
II	109160.7978	109161.0138	0.000%	109160.7978	0.000%
III	10009990.0200	10010041.3789	0.000%	10009990.0200	0.000%
IV	37015.6212	37016.0426	0.000%	37015.6212	0.000%
V	287569.3725	287569.4357	0.000%	287569.3725	0.000%
VI	16180.3399	16180.4065	0.000%	16180.3399	0.000%
VII	16180.3399	16180.3784	0.000%	16180.3399	0.000%
VIII	10000.5000	10000.5000	0.000%	10000.5000	0.000%
IX	431004.0987	431004.4182	0.000%	431004.0987	0.000%
X	10000.9999	10001.0038	0.000%	10000.9999	0.000%

Table 6.5. Comparison of solutions for the linear-quadratic problem

N	Exact solution	GAMS value	GAMS D	GAMS+ value	GAMS+ D	Genetic Alg value	Genetic Alg D
2	6.331738	4.3693	30.99%	6.3316	0.00%	6.3317	0.000%
4	12.721038	5.9050	53.58%	12.7210	0.00%	12.7210	0.000%
8	25.905710	*		18.8604	27.20%	25.9057	0.000%
10	32.820943	*		22.9416	30.10%	32.8209	0.000%
20	73.237681	*		*		73.2376	0.000%
45	279.275275	*		*		279.2714	0.001%

Table 6.6. Comparison of solutions for the harvest problem. The symbol * means that the GAMS failed to report a reasonable value

GA vs. GAMS (General Algebraic Modeling System)

- Time

N	No. of iterations needed	Time needed (CPU sec)	Time for 40,000 iterations (CPU sec)	Time for GAMS (CPU sec)
5	6234	65.4	328.9	31.5
10	10231	109.7	400.9	33.1
15	19256	230.8	459.8	36.6
20	19993	257.8	590.8	41.1
25	18804	301.3	640.4	47.7
30	22976	389.5	701.9	58.2
35	23768	413.6	779.5	68.0
40	25634	467.8	850.7	81.3
45	28756	615.9	936.3	95.9

Table 6.8. Time performance of evolution program and GAMS for the push-cart problem (6.9)–(6.11): number of iterations needed to obtain the result with precision of six decimal places, time needed for that number of iterations, time needed for all 40,000 iterations

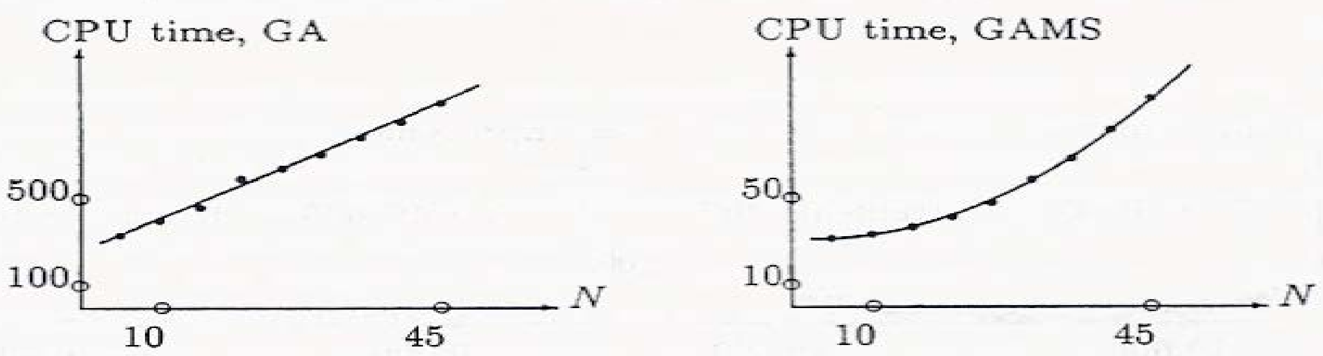

Fig. 6.2. Time as a function of problem size (N).

Crossover

- Simple crossover

$$v_1 = (x_1, x_2, x_3, x_4) \qquad v'_1 = (y_1, x_2, x_3, x_4)$$
$$v_2 = (y_1, y_2, y_3, y_4) \qquad v'_2 = (x_1, y_2, y_3, y_4)$$

- Uniform crossover*

$$v_1 = (x_1, x_2, x_3, x_4) \qquad v'_1 = (x_1, x_2, x_3, y_4)$$
$$v_2 = (y_1, y_2, y_3, y_4) \qquad v'_2 = (y_1, y_2, y_3, x_4)$$

- Arithmetical crossover (simple*/single*/whole*)

$$x_i^{t+1} = a \cdot x_i^t + (1-a) \cdot y_i^t$$
$$y_i^{t+1} = a \cdot y_i^t + (1-a) \cdot x_i^t$$

where $a \in [0..1]$; a *is constant*.

- Non-uniform arithmetical crossover

$$x_i^{t+1} = a^t \cdot x_i^t + (1-a^t) \cdot y_i^t$$
$$y_i^{t+1} = a^t \cdot y_i^t + (1-a^t) \cdot x_i^t$$

where $a^t \in [0..1]$; a^t *is variable*.

Adaptive Probabilities (P_c and P_m)

$$p_c = \begin{cases} k_1 \cdot (f_{\max} - f') \big/ (f_{\max} - \bar{f}) & \text{if} \quad f' \geq \bar{f} \\ k_3 & \text{otherwise} \end{cases}$$

$$\textit{where} \quad f' = \max\{f(x), f(y)\}$$

$$p_m = \begin{cases} k_2 \cdot (f_{\max} - f) \big/ (f_{\max} - \bar{f}) & \text{if} \quad f \geq \bar{f} \\ k_4 & \text{otherwise} \end{cases}$$

$$\textit{where} \quad 0 \leq k_i \leq 1$$