Multi-Objective Evolutionary Algorithms for Resource Allocation Problems

Dilip Datta

(ddatta@iitk.ac.in)

Supervisors:

Dr. Carlos M. Fonseca	**Dr. Kalyanmoy Deb**
Professor Auxiliar, DEEI, FST	Professor, Mechanical Engineering
University of Algarve, Portugal	IIT-Kanpur, India
(cmfonsec@ualg.pt)	(deb@iitk.ac.in)

Organization of the Thesis

- Introduction to Resource Allocation Problems (RAPs)
- Aim of the thesis
- Study of two RAPs
  - Introduction
  - Literature review
  - Formulation as multi-objective optimization problems
  - Design of evolutionary algorithms
  - Case studies
- Similarity study between the RAPs
- Conclusions and future research scope

Resource Allocation Problem (RAP)

- It is encountered in operations research and management science, including load distribution, production planning, computer scheduling, portfolio selection, apportionment, etc.
- It is an optimization problem where limited amount of resources are to be allocated to certain number of competitive events/activities in order to achieve the most effective allotment of resources.
  - **Objectives**: usually minimization of overall cost, processing time, or conflicts; or maximization of net profit, efficiency, or compatibility.

Nature of an RAP

- An RAP is an integer or mixed-integer programming problem, in particular a combinatorial optimization problem.
  - It usually contains huge number of integer and/or real variables and constraints, a discrete and finite search space, and multiple objectives.
- The worst case complexity of an RAP is NP-complete (Ibaraki and Katoh [1988], Zhang [2002]).
- It is believed that the underlying properties of an NP-complete RAP can help in characterizing another NP-complete RAP (Zhang [2002]).

Aim of the Present Work

- To study the follwing two RAPs of quite different natures:
  - University class timetabling problem, and
  - Land-use management problem.
- Finally to find the similarities between the problems, and also between their solution techniques.

University Class Timetabling Problem

Class Timetabling Problem

- It involves the scheduling of classes (lectures), students, teachers and rooms at a fixed number of time-slots.
- It has to respect certain restrictions:
  - No class, student, teacher or room can be engaged more than once at a time, and many more.
- An effective timetable is crucial for the satisfaction of educational requirements, and the efficient utilization of human and space resources.

Literature Review on Class Timetabling Problem

- It is generally tackled as a single-objective optimization problem. Only in recent years, a few multi-objective optimization techniques have been proposed.
- Most of the works were concentrated on school timetabling, and only a few on university class timetabling.
- On the other hand, most of the works were based on simple class-structures. Only a limited number of researchers considered one or more complex class-structures, e.g., multi-slot or combined classes.
- Classical methods are not fully capable to handle this highly constrained combinatorial problem, and evolutionary algorithms are the most widely used non-classical techniques.

Aim of the Present Work on Class Timetabling Problem

- To study different variants of the problem.
- To formulate a model university class timetabling problem, considering
  - Different class-structures,
  - Constraints that are common to most of its variants, and
  - Multiple objectives to evaluate the quality of a solution.
- To develop a multi-objective evolutionary algorithm for it.
- Case studies.

Types of Academic Courses

- Simple Course
- Compound Course
  - Multi-Slot Class
  - Split Class
  - Combined Class
- Open Course (Optional Course)
  - Group Courses

Common Hard Constraints

1. A student should have only one class at a time.
2. A teacher should have only one class at a time.
3. A room should be booked only for one class at a time (a set of combined classes may be treated as a single class).
4. A course should have only one class on a day.
5. A class should be scheduled only in a specific room, if required, otherwise in any room which has sufficient sitting capacity for the students of the class.
6. A class should be scheduled only at a specific time-slot, if required.

Common Soft Constraints

1. A student should not have any free time-slot between two classes on a day.
2. A teacher should not be booked at consecutive time-slots on a day (except in multi-slot classes).
3. A smaller class should not be scheduled in a room which can be used for a bigger class.

Common Objective Functions

1. Minimize the average number of weekly free time-slots between two classes of a student, and
2. Minimize the average number of weekly consecutive classes of a teacher.

Mathematical Formulation

- Objective functions:
  1. Minimize the average number of weekly free time-slots between two classes of a student:

$$f_1 \equiv \frac{1}{\mathrm{S}} \sum_{s=1}^{\mathrm{S}} \sum_{i=1}^{\mathrm{D}} \sum_{j=t^{\mathrm{o}}_{s,i}}^{\mathrm{t}^{\mathrm{e}}_{\mathrm{s,i}}} I\left(\sum_{e=1}^{\mathrm{E}} d_{e,i}.t_{e,j}.p_{e,s} = 0 \right) \quad , \qquad (1)$$

  2. Minimize the average number of weekly consecutive classes of a teacher:

$$f_2 \equiv \frac{1}{\mathrm{M}} \sum_{m=1}^{\mathrm{M}} \sum_{i=1}^{\mathrm{D}} \sum_{j=1}^{\mathrm{T}} I_1(A_1 > 0).I_2(A_2 > 0).I_3(A_3 = 0) \quad , \qquad (2)$$

  where $A_1 = \sum_{e=1}^{\mathrm{E}} d_{e,i}.t_{e,j-1}.\mu_{e,m}$, $A_2 = \sum_{e=1}^{\mathrm{E}} d_{e,i}.t_{e,j}.\mu_{e,m}$ and $A_3 = \sum_{e=1}^{\mathrm{E}} d_{e,i}.t_{e,j-1}.t_{e,j}.\mu_{e,m}$.

Mathematical Formulation (Contd...)

- Subject to hard constraints:

1. Number of classes to be attended by a student at a time:

$$g_{(s-1)\mathrm{TD}+(i-1)\mathrm{T}+j} \equiv \sum_{e=1}^{\mathrm{E}} d_{e,i}.t_{e,j}.p_{e,s} \le 1, \qquad (3)$$
$$s = 1, .., \mathrm{S};\; i = 1, .., \mathrm{D}; \;\text{ and } j = 1, .., \mathrm{T} \;,$$

2. Number of classes to be taught by a teacher at a time:

$$g_{\mathrm{STD}+(m-1)\mathrm{TD}+(i-1)\mathrm{T}+j} \equiv \sum_{e=1}^{\mathrm{E}} d_{e,i}.t_{e,j}.\mu_{e,m} \le 1, \qquad (4)$$
$$m = 1, .., \mathrm{M};\; i = 1, .., \mathrm{D}; \;\text{ and } j = 1, .., \mathrm{T} \;,$$

3. Number of classes in a room at a time:

$$g_{(\mathrm{S}+\mathrm{M})\mathrm{TD}+(k-1)\mathrm{TD}+(i-1)\mathrm{T}+j} \equiv \sum_{e=1}^{\mathrm{E}} d_{e,i}.t_{e,j}.r_{e,k} \le 1, \qquad (5)$$
$$i = 1, .., \mathrm{D};\; j = 1, .., \mathrm{T}; \;\text{ and } k = 1, .., \mathrm{R} \;,$$

4. Number of classes of a course on a day:

$$g_{(\mathrm{S+M+R})\mathrm{TD}+(c-1)\mathrm{D}+i} \equiv \sum_{e=1}^{\mathrm{E}} d_{e,i}.l_{e,c} \leq 1, \quad c = 1,..,\mathrm{C}; \text{ and } i = 1,..,\mathrm{D} \ , \tag{6}$$

5. Room where a class is to be scheduled:
   (a) Type of the room:

$$g_{(\mathrm{S+M+R})\mathrm{TD+CD}+2e-1} \equiv \sum_{k=1}^{\mathrm{R}} r_{e,k}.I(k \in \rho_e) = 1, \quad e = 1,..,\mathrm{E} \ , \tag{7}$$

   (b) Capacity of the room:

$$g_{(\mathrm{S+M+R})\mathrm{TD+CD}+2e} \equiv \sum_{s=1}^{\mathrm{S}} p_{e,s} \leq \sum_{k=1}^{\mathrm{R}} r_{e,k}.h_k, \quad e = 1,..,\mathrm{E} \ , \tag{8}$$

6. Time-slot at which a class is to be scheduled:

$$g_{(\mathrm{S+M+R})\mathrm{TD+CD}+2E+e} \equiv \sum_{j=1}^{\mathrm{T}} t_{e,j}.I(j \in \tau_e) = 1, \quad e = 1,..,\mathrm{E} \ , \tag{9}$$

Mathematical Formulation (Contd...)

- Soft constraints:
  1. Total number of weekly free time-slots between two classes of students:

$$sc_1 \equiv \sum_{s=1}^{\mathrm{S}} \sum_{i=1}^{\mathrm{D}} \sum_{j=t^{\mathrm{o}}_{s,i}}^{\mathrm{t^e_{s,i}}} I\left(\sum_{e=1}^{\mathrm{E}} d_{e,i}.t_{e,j}.p_{e,s} = 0\right) = 0 \ , \qquad (10)$$

  2. Total number of weekly consecutive classes teachers:

$$sc_2 \equiv \sum_{m=1}^{\mathrm{M}} \sum_{i=1}^{\mathrm{D}} \sum_{j=1}^{\mathrm{T}} I_1(A_1 > 0).I_2(A_2 > 0).I_3(A_3 = 0) = 0 \ , \qquad (11)$$

  3. Capacity of a class, and that of the room where the class is scheduled:

$$sc_{2+\sum_{e''=0}^{e-1} \mathrm{K}_{e''}+(e'-e)} \equiv \sum_{s=1}^{\mathrm{S}} p_{e',s} > \sum_{k=1}^{\mathrm{R}} r_{e,k}.h_k, \text{ if } \sum_{s=1}^{\mathrm{S}} p_{e',s} > \sum_{s=1}^{\mathrm{S}} p_{e,s} \ , \qquad (12)$$

     where $\mathrm{K}_{e''}$=0 if e''=0, else $\mathrm{K}_{e''}$=E-e'', and e=1,..,E-1 and e'=e+1,..,E.

Mathematical Formulation (Contd...)

1. Number of objective functions : 2
2. Number of hard constraints : (S+M+R)TD+CD+3E
3. Number of soft constraints : $2 + \frac{E(E-1)}{2}$

where,
- S = Number of students,
- M = Number of teachers,
- R = Number of rooms,
- T = Number of time-slots/day,
- D = Number of days/week,
- C = Number of courses, and
- E = Number of events (classes)

Heuristic Approach for Generating Initial Solutions

- All classes are first sorted in the following order:
  1. Ascending order of number of specific time-slots,
  2. Else, descending order of number of time-slots/class,
  3. Else, ascending order of number of specific rooms,
  4. Else, preference to group classes,
  5. Else, preference to split classes,
- Assign time-slots and rooms to the classes taking in order.

Characteristics of Class Timetabling Problem

1. Random scheduling of events may lead to the improper use of resources, and as a result, many later events may be left unscheduled due to the non-availability of proper resources.

2. Even if the events are scheduled in order, many resources may be required to leave unused, i.e., sufficient events may not be available to use all the resources.

3. Amount of unused resources increases with the increasing complexities of the events.

Chromosome Representation

A direct chromosome/solution representation has been used:

- A chromosome is a two-dimensional matrix, each column of which represents a time-slot and each row represents a room.
  - That is, a chromosome is a vector of time-slots, and a time-slot (gene) is a vector of rooms.
  - Each cell of the matrix represents class(es) that is(are) scheduled in the corresponding room and time-slot.

Chromosome Representation (Contd...)

Table 1: Chromosome representation.

R/T	$\mathbf{T}_1$	$\mathbf{T}_2$	$\mathbf{T}_3$	..	$\mathbf{T}_j$	..	$\mathbf{T}_t$
$\mathbf{R}_1$	C20	C11	C39	...	C05	...	C16
$\mathbf{R}_2$	C33	C21	C15	...	C40	...	C12
$\mathbf{R}_3$	C01	C35		...	C07	...	C08
..	...	...	...	...	...	...	C27
$\mathbf{R}_k$	C13	C02	C14	...	C22	...	C38
..	...	...	...	...	...	...	C18
$\mathbf{R}_r$	C06	C04	C17 C36	...	C28	...	C31

Traditional Crossover Operator

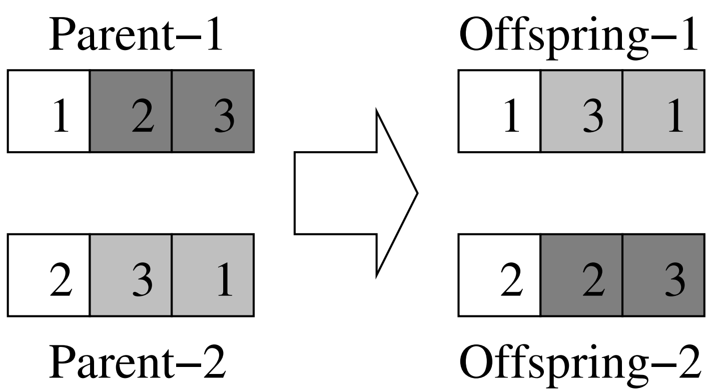


Fig. 1: Traditional single-point crossover operator.

Crossover for Valid Resource Allocation (XVRA)

First parent

	MON		TUE		WED		THU		FRI	
	T1	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**	**T10**
R1	*C1*			*C3*	*C5*		*C9*			
R2						*C4*		*C7*	*C2*	*C6*
R3		*C10*	*C8*							

Second parent

	MON		TUE		WED		THU		FRI	
	T1	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**	**T10**
R1		*C2*		*C7*		*C1*			*C9*	
R2	*C6*		*C8*				*C5*	*C3*		*C4*
R3					*C10*					

	MON		TUE		WED		THU		FRI	
	T1	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**	**T10**
R1	*C1*			*C7*	*C5*				*C9*	
R2			*C8*			*C4*				
R3		*C10*								

First offspring generated from first parent with the help of second parent

	MON		TUE		WED		THU		FRI	
	T1	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**	**T10**
R1	*C1*			*C7*	*C5*				*C9*	
R2			*C8*			*C4*		*C6*		*C3*
R3		*C10*					*C2*			

First offspring completed by heuristic approach

Fig. 2: Generation of the first offspring using XVRA.

Mutation Operators

1. MFRA: Mutation for Fresh Resource Allocation
2. MIRA: Mutation for Interchanging Resource Allocation
3. MRRA: Mutation for Reshuffling Resource Allocation
4. MSIS: Mutation for Steering Infeasible Solution

Mutation Operator: MRRA

	MON		TUE		WED		THU		FRI	
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
R1		C10	C7				C6		C10	
R2			C8	C8	C7					C6
R3	C4			C3					C3	
R4			C1			C9			C5	C4
R5	C5		C2						C9	

Chosen Time-slot (T3) Chosen Time-slot (T9)

(a) Before mutation

	MON		TUE		WED		THU		FRI	
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
R1		C10	C10				C6		C7	
R2				C6	C7				C8	C8
R3				C3				C4	C3	
R4			C5			C9			C1	C4
R5	C5		C9						C2	

Chosen Time-slot (T3) Chosen Time-slot (T9)

(b) After mutation

Fig. 3: Mutation for Reshuffling Resource Allocation (MRRA).

Mutation Operator: MSIS

	MON		TUE		WED		THU		FRI	
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
R1	*C1*			*C7*	*C5*		*C2*			
R2			*C8*			*C4*		*C6*	*C8*	
R3		*C1*							*C9*	

(a) Before mutation

	MON		TUE		WED		THU		FRI	
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
R1	*C2*			*C7*	*C5*		*C1*			
R2			*C8*		*C8*	*C4*		*C6*		
R3		*C1*							*C9*	

(b) After mutation

Fig. 4: Mutation for steering infeasible solution towards feasible region.

NSGA-II-UCTO: NSGA-II as University Class Timetable Optimizer

1. Provisions for availing through input data files only:
   (a) Compound courses, having multi-slot, split and combined classes.
   (b) Open courses, including group courses.
   (c) Choices for specific rooms and time-slots.
   (d) Status of constraint handling.
2. Addition/deletion of any constraint through subroutines.
3. It can also be used to school timetabling problem with little modification in objective and constraint functions.

Application of NSGA-II-UCTO

1. **NITS1:** Odd-semester class timetable of NIT-Silchar
2. **NITS2:** Even-semester class timetable of NIT-Silchar
3. **IITK2:** Even-semester class timetable for common compulsory classes of B.Tech and integrated M.Sc of IIT-Kanpur, including slots for departmental compulsory and elective classes of Mechanical Engineering department for its B.Tech programme.

NITS1: Odd-Semester Classes of NIT-Silchar

Table 2: Classes in NITS1.

Classes	Number of Classes				No. of Slots
	Simple	Split	Combined	Open/Group	
1-Slot	254	98	36	62	432
2-Slot	15	50	–	–	130
3-Slot	05	02	–	–	21
Total	**274**	**150**	**36**	**62**	**583**

Total 522 classes spanning over 583 slots

NITS2: Even-Semester Classes of NIT-Silchar

Table 3: Classes in NITS2.

Classes	Number of Classes				No. of Slots
	Simple	Split	Combined	Open/Group	
1-Slot	238	98	42	66	423
2-Slot	06	46	–	–	104
3-Slot	–	08	–	–	24
4-Slot	–	02	–	–	08
Total	**244**	**154**	**42**	**66**	**559**

Total 506 classes spanning over 559 slots

IITK2: Even-Semester Classes of IIT-Kanpur

Table 4: Common compulsory classes of IITK2.

Classes	**Number of Classes**				**No. of Slots**
	Simple	**Split**	**Combined**	**Open/Group**	
1-Slot	11	–	–	219	230
3-Slot	–	12	–	–	36
Total	**11**	**12**	–	**219**	**266**

Total 242 classes spanning over 266 slots

Hard constraints in NITS1, NITS2 and IITK2

Table 5: Numbers of hard constraints in NITS1, NITS2 and IITK2.

Prob.	S	M	R	T	D	C	E	No. of Constraints (S+M+R)TD+CD+3E
NITS1	860	89	37	7	5	154	522	36,846
NITS2	860	63	35	7	5	143	506	35,763
IITK2	1000	103	40	8	5	125	242	47,071

Replacing student-clash constraints by batch-clash constraints, the number of hard constraints in NITS1 and IITK2 could be reduced as below:

- In NITS1, from 36,846 to 7,411.
- In IITK2, from 47,071 to 7,151.

Selection of Mutation Operators

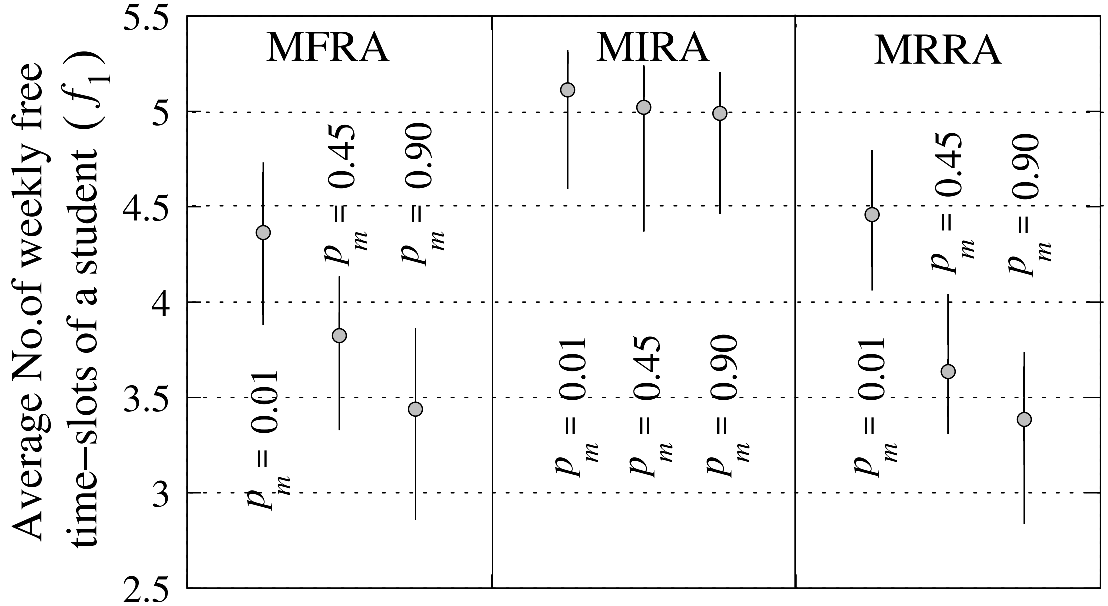


Fig. 5: Performance of MFRA, MIRA and MRRA on objective function f_1 of NITS2.

Selection of Mutation Operators (Contd...)

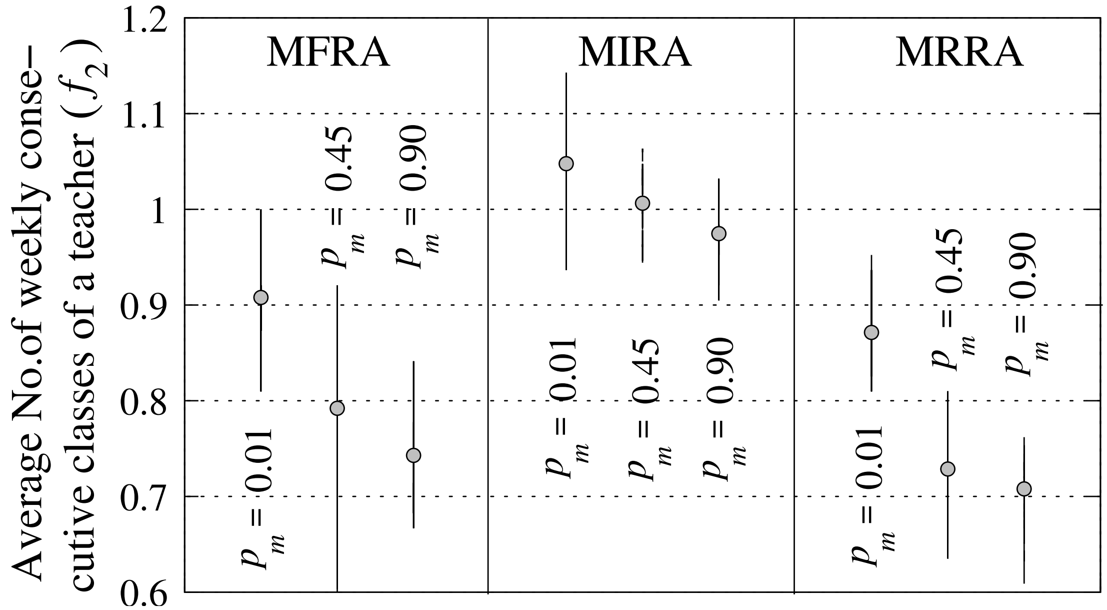
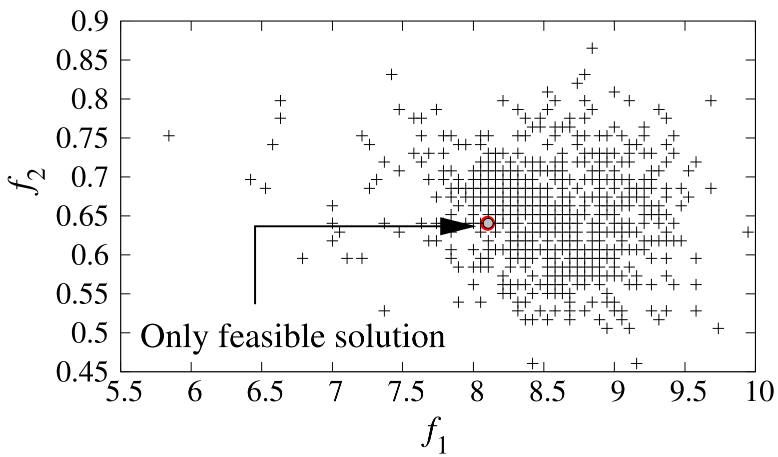
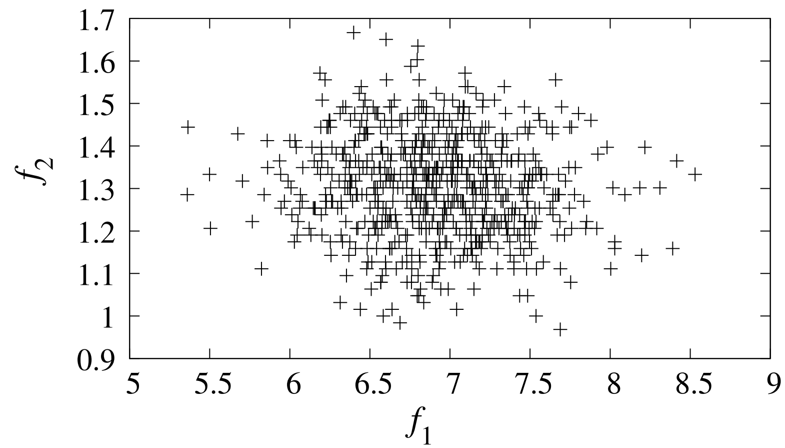


Fig. 6: Performance of MFRA, MIRA and MRRA on objective function f_2 of NITS2.

NITS1 and NITS2 with Relaxed Hard Constraints



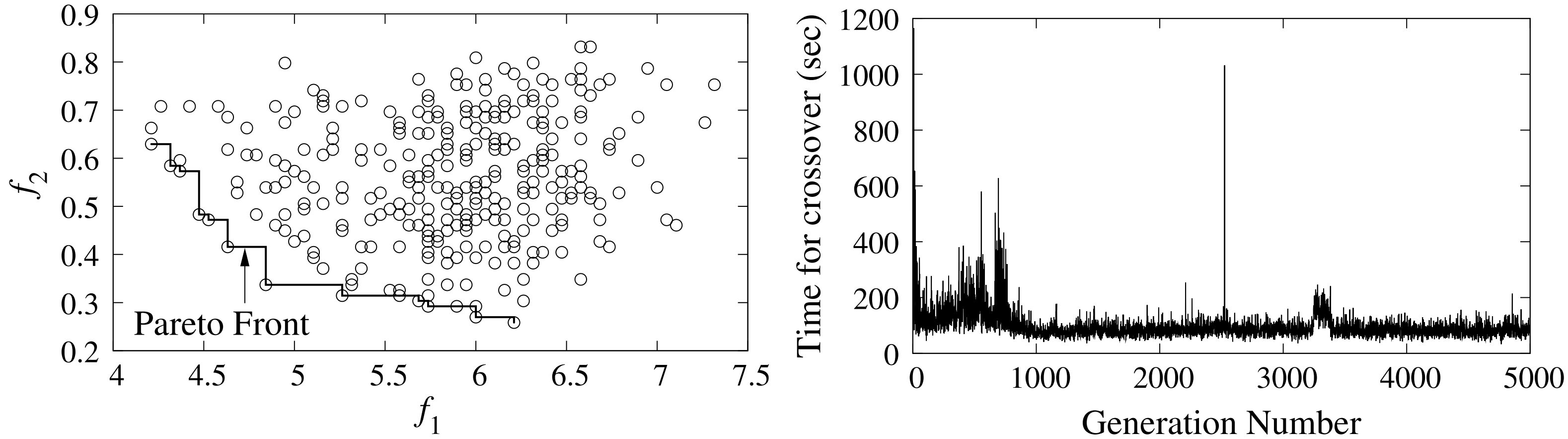
(a) NITS1 (only one feasible solution)



(b) NITS2 (no feasible solution)

Fig. 7: NITS1 and NITS2 with relaxed student and teacher-clash constraints ($p_c = 0.90$, $p_m = 0.01$ and $r_s = 0.125$).

NITS1 Maintaining Feasibility of Solutions

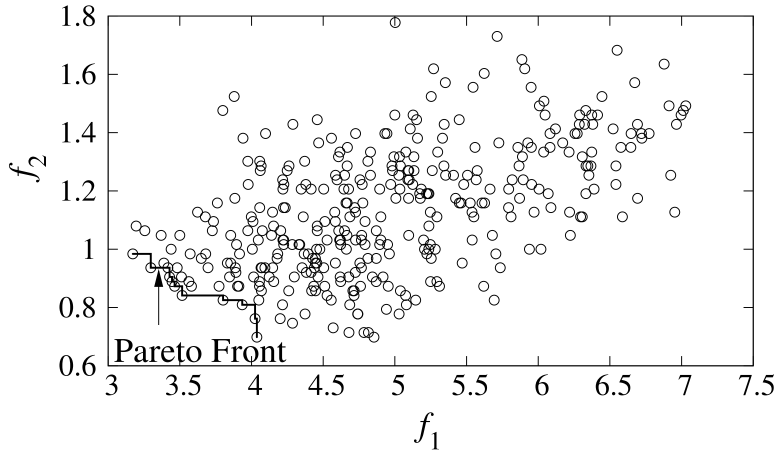


(a) Final solutions (b) Time for crossover

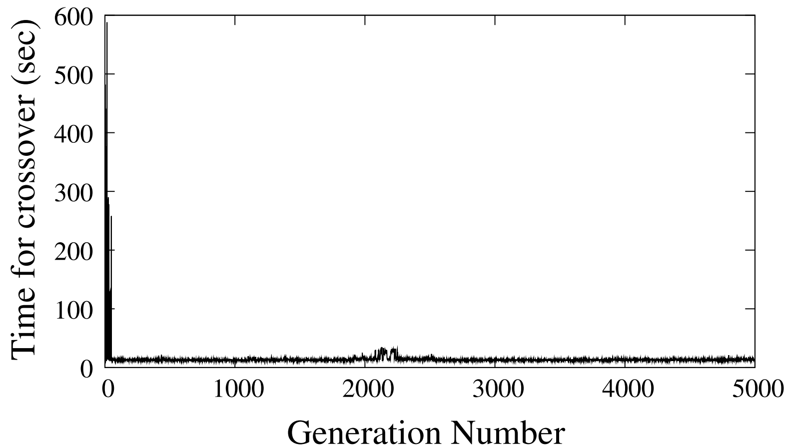
Fig. 8: Solutions of NITS1 ($p_c = 0.90$, $p_m = 0.01$ and $r_s = 0.125$).

Total execution time = 138 hours 32 minutes 51 seconds.

NITS2 Maintaining Feasibility of Solutions



(a) Final solutions

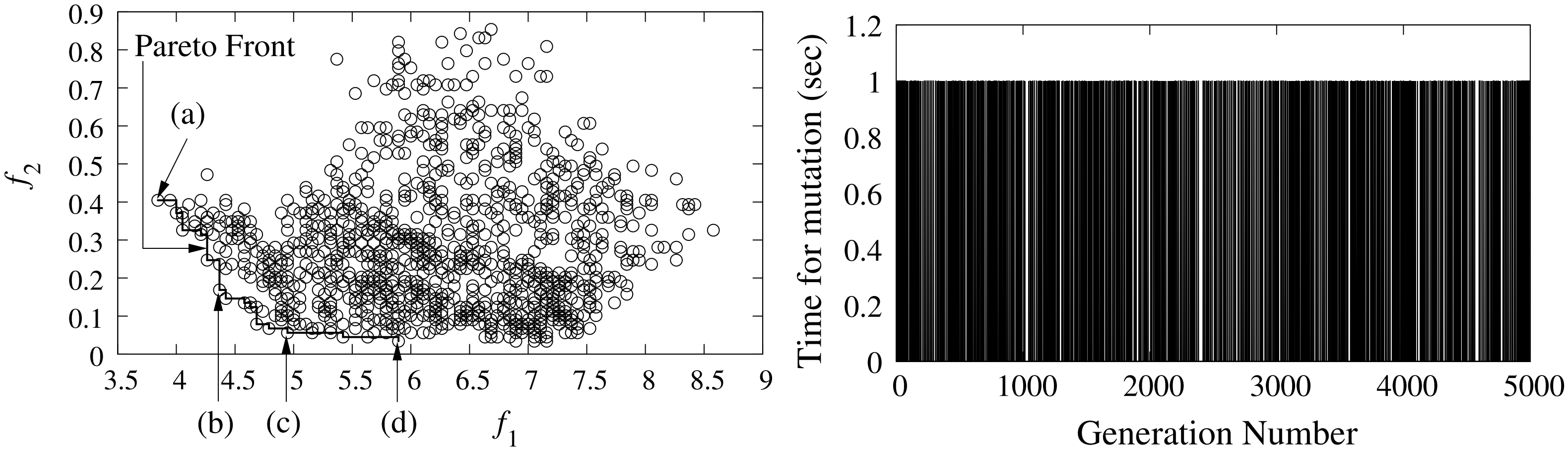


(b) Time for crossover

Fig. 9: Solutions of NITS2 ($p_c = 0.90$, $p_m = 0.01$ and $r_s = 0.125$).

Total execution time = 22 hours 15 minutes 33 seconds.

NITS1 Using Only Mutation (MRRA)



(a) Final solutions (b) Time for mutation

Fig. 10: Solutions of NITS1 using only MRRA ($p_m = 0.90$ and $r_s = 0.125$).

Total execution time = 1 hour 19 minutes 17 seconds.

NITS1 Using Only Mutation (MRRA) (Contd...)

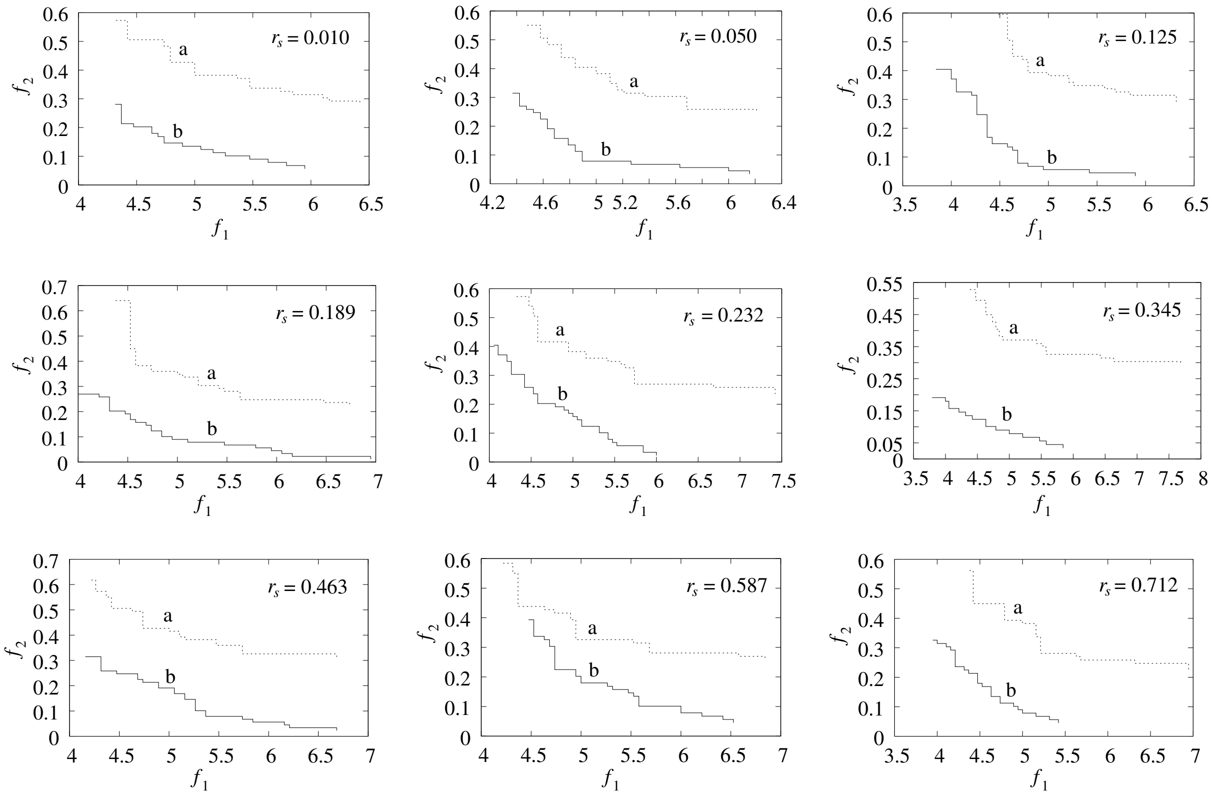
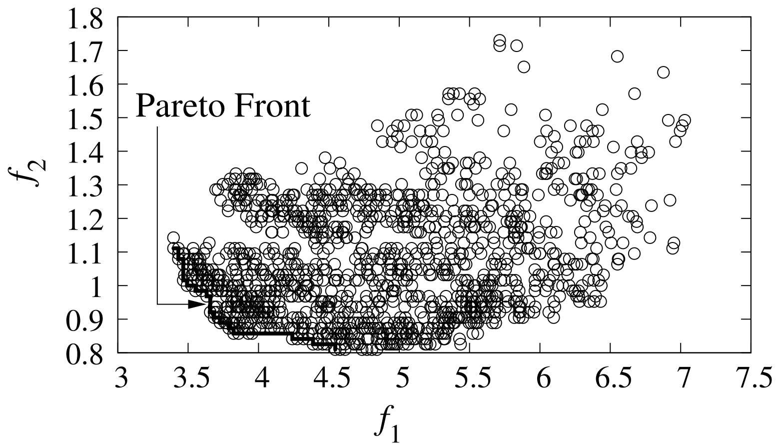
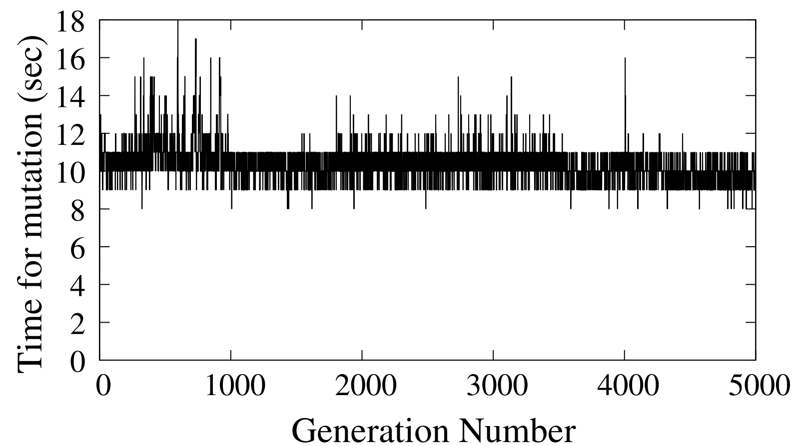


Fig. 11: Comparison of Pareto fronts of NITS1. Curve (a): combined XVRA and MRRA, and curve (b): only MRRA.

NITS2 Using Only Mutation (MRRA)



(a) Final solutions



(b) Time for mutation

Fig. 12: Solutions of NITS2 using only MRRA ($p_m = 0.90$ and $r_s = 0.125$).

Total execution time = 15 hours 58 minutes 14 seconds.

NITS2 Using Only Mutation (MRRA) (Contd...)

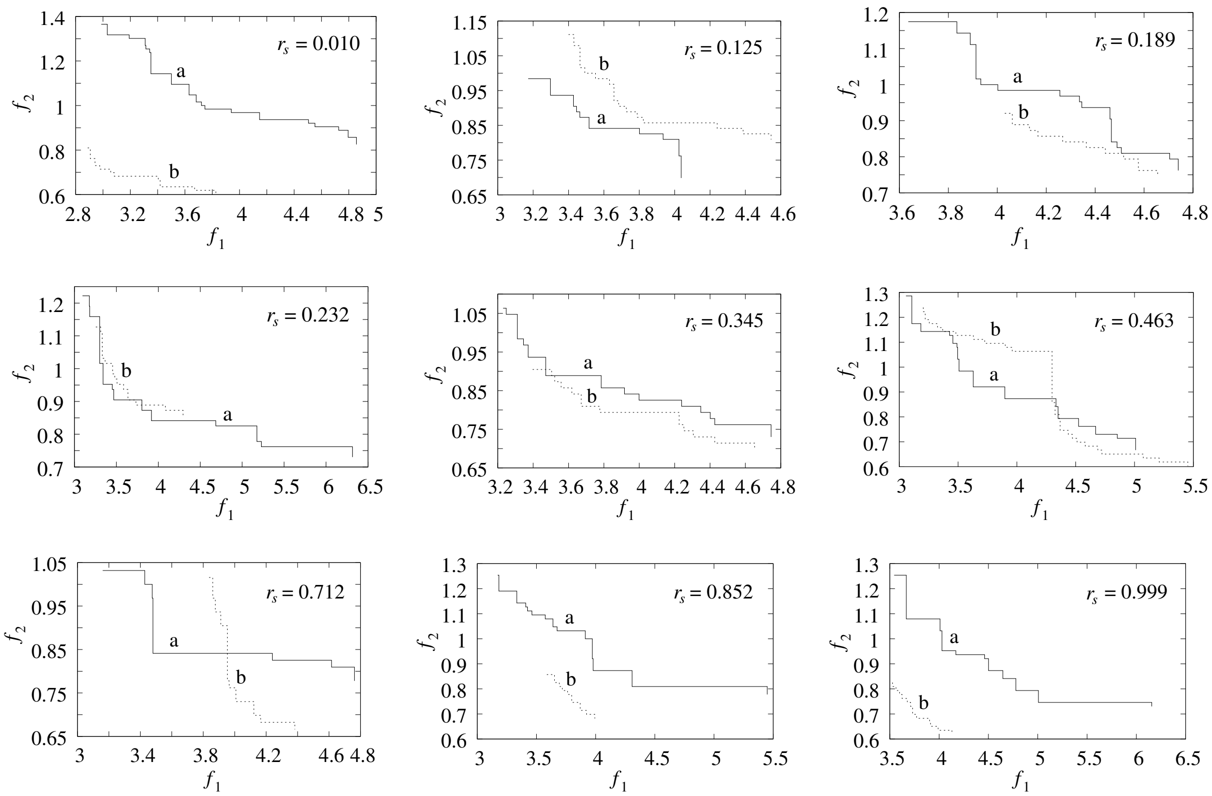
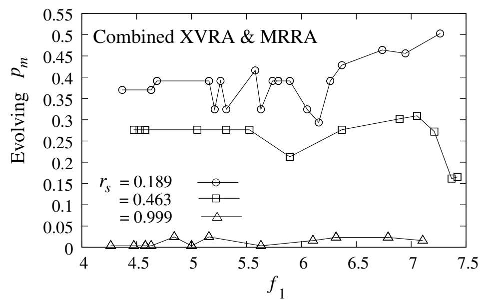
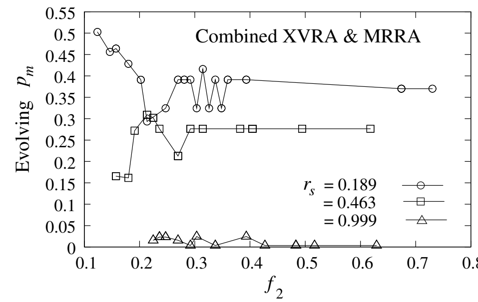


Fig. 13: Comparison of Pareto fronts of NITS2. Curve (a): combined XVRA and MRRA, and curve (b): only MRRA.

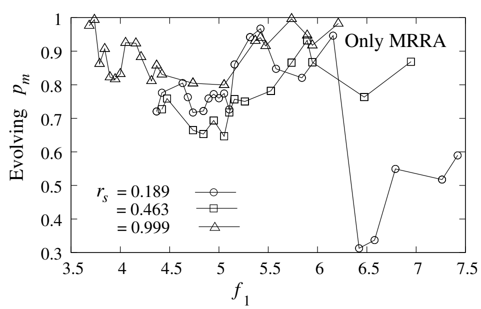
Evolving Mutation Probabilities in NITS1



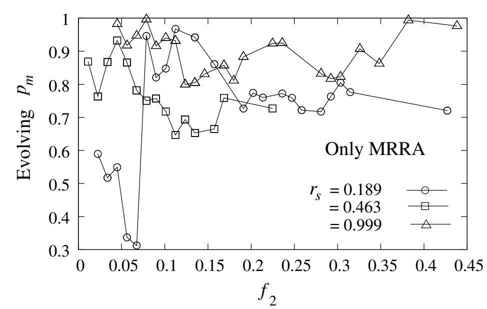
(a) Combined XVRA and MRRA



(b) Combined XVRA and MRRA



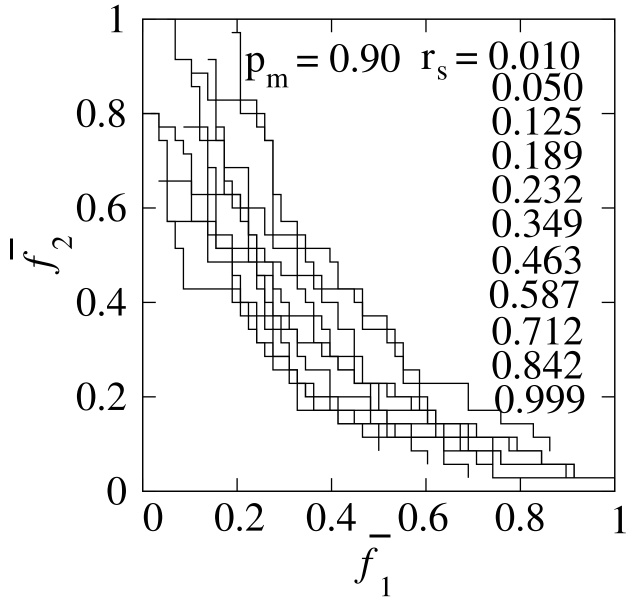
(c) Only MRRA



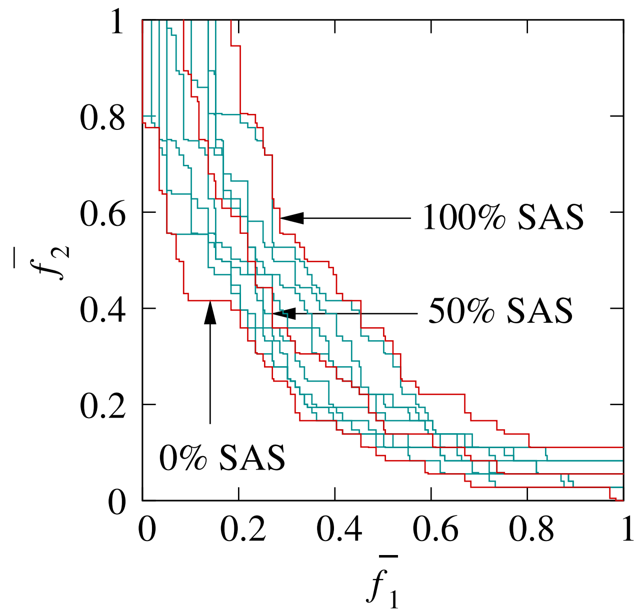
(d) Only MRRA

Fig. 14: Evolving p_m against objective functions of NITS1.

Attainment and Summary Attainment Surfaces of NITS1



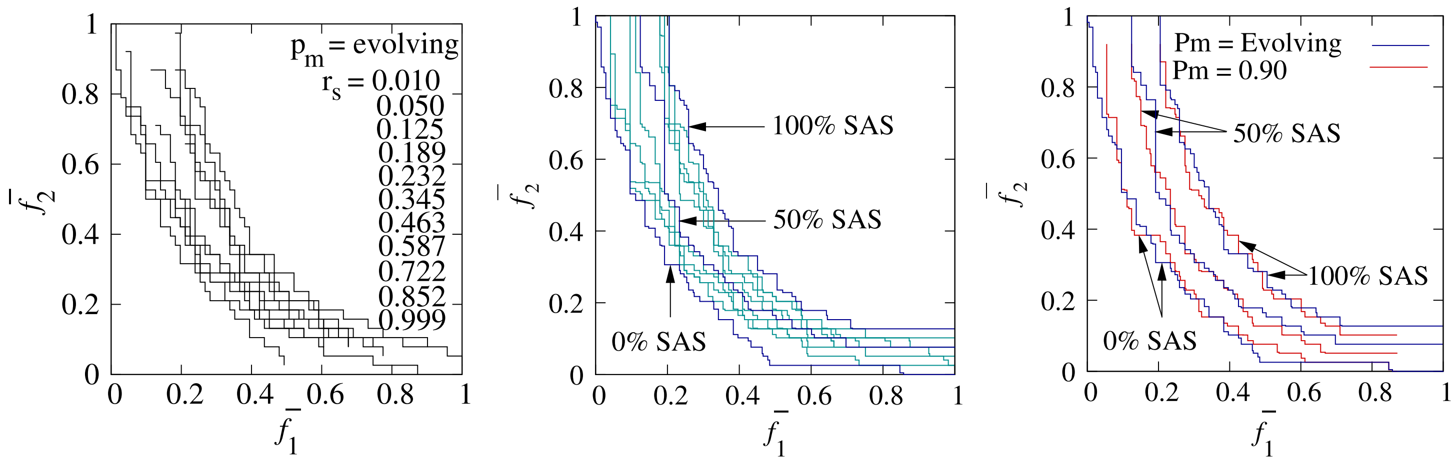
(a) AS



(b) SAS

Fig. 15: AS and SAS for NITS1 using only MRRA with $p_m = 0.90$.

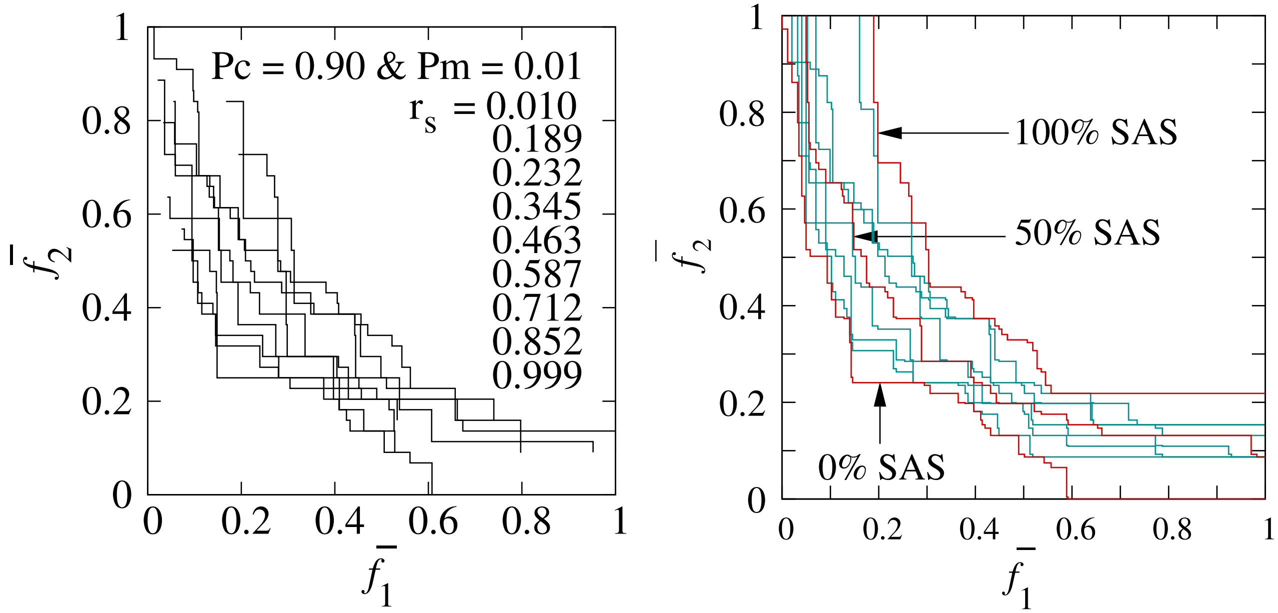
Attainment and Summary Attainment Surfaces of NITS1 (Contd...)



(a) AS (b) SAS (c) Comparison SAS

Fig. 16: AS and SAS for NITS1 using only MRRA with evolving p_m, and comparison of SAS with those shown in Fig.15.

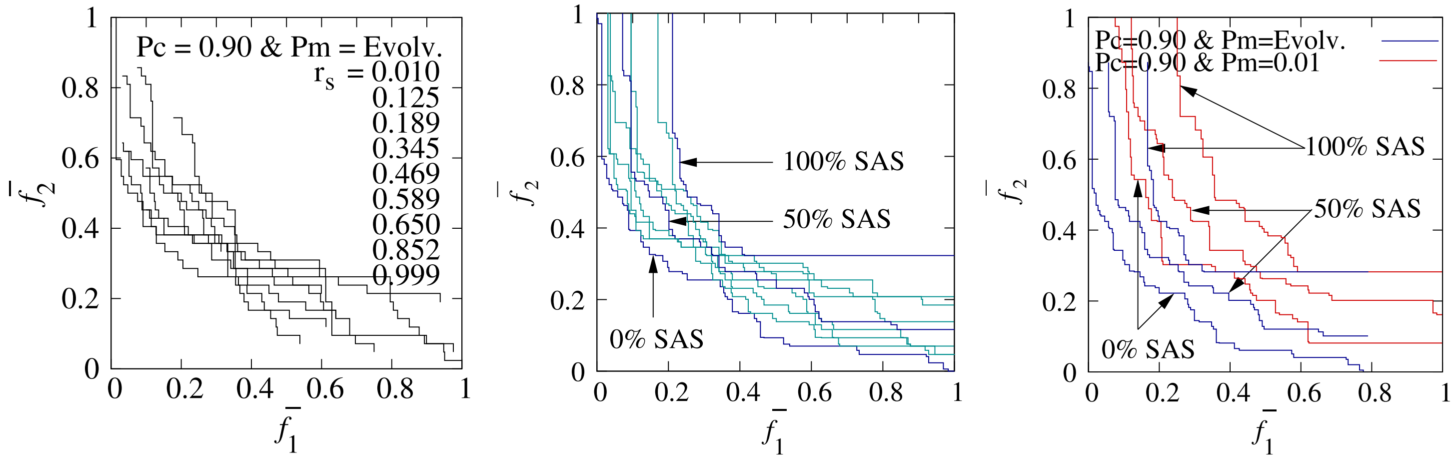
Attainment and Summary Attainment Surfaces of NITS2



(a) AS (b) SAS

Fig. 17: AS and SAS for NITS2 using combined XVRA and MRRA with $p_c = 0.90$ and $p_m = 0.010$.

Attainment and Summary Attainment Surfaces of NITS2 (Contd...)



(a) AS (b) SAS (c) Comparison SAS

Fig. 18: AS and SAS for NITS2 using XVRA and MRRA with $p_c = 0.90$ and evolving p_m, and comparison of SAS with those shown in Fig.17.

Comparison of Results of NITS1 under Single and Multi-Objective Optimization

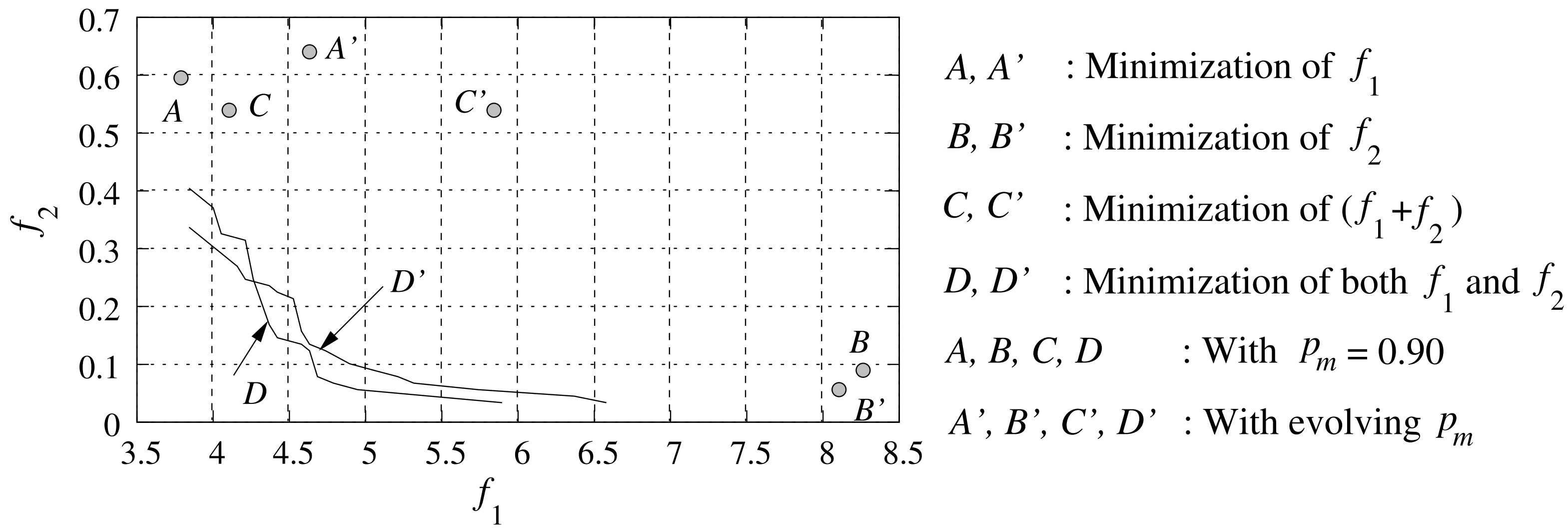


Fig. 19: Solutions of NITS1 under single and multi-objective optimization.

Existing Solution of IITK2

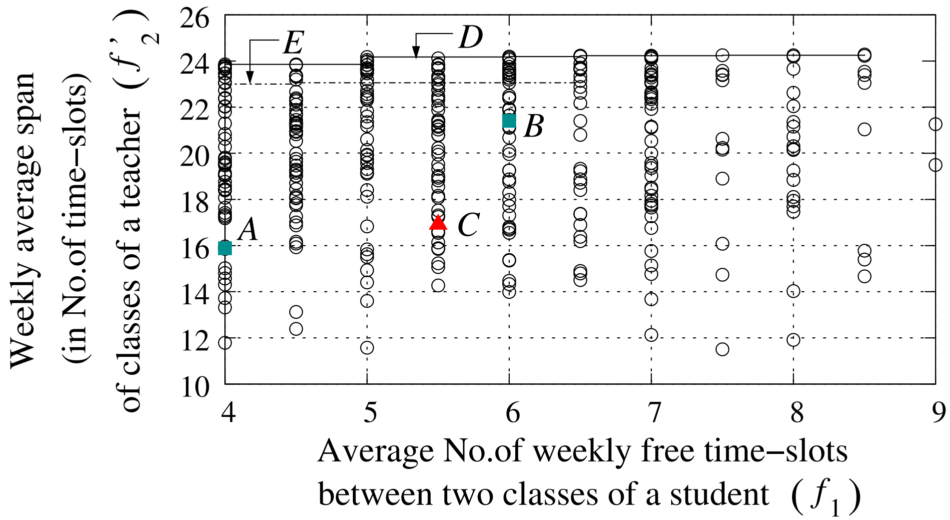
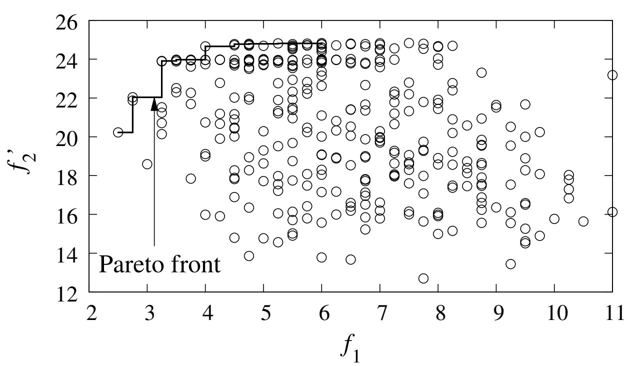
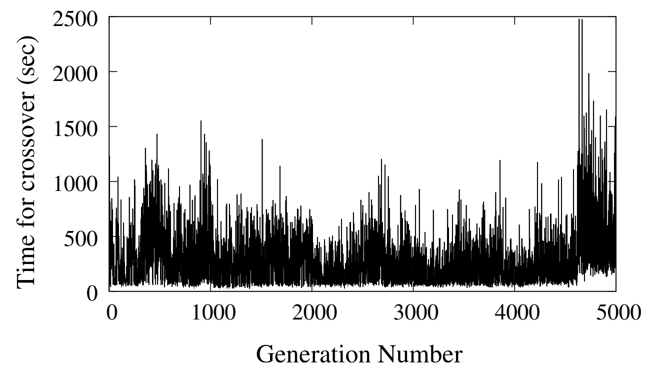


Fig. 20: Scheduling of common compulsory classes of IITK2. A and B: Single-objective optimization of f_1 and f'_2; C: Manually prepared timetable which is in use; D: Multi-objective optimization with combined XVRA and MRRA, and E: Multi-objective optimization with MRRA alone.

IITK2 under Combined XVRA and MRRA



(a) Final solutions



(b) Time for mutation

Fig. 21: Solutions of IITK2 using combined XVRA and MRRA.

- Total execution time = 465 hours 14 minutes 39 seconds.
- When solved under MRRA alone, this time was reduced to 11 hours 31 minutes 42 seconds.

Comparison of Pareto Fronts of IITK2

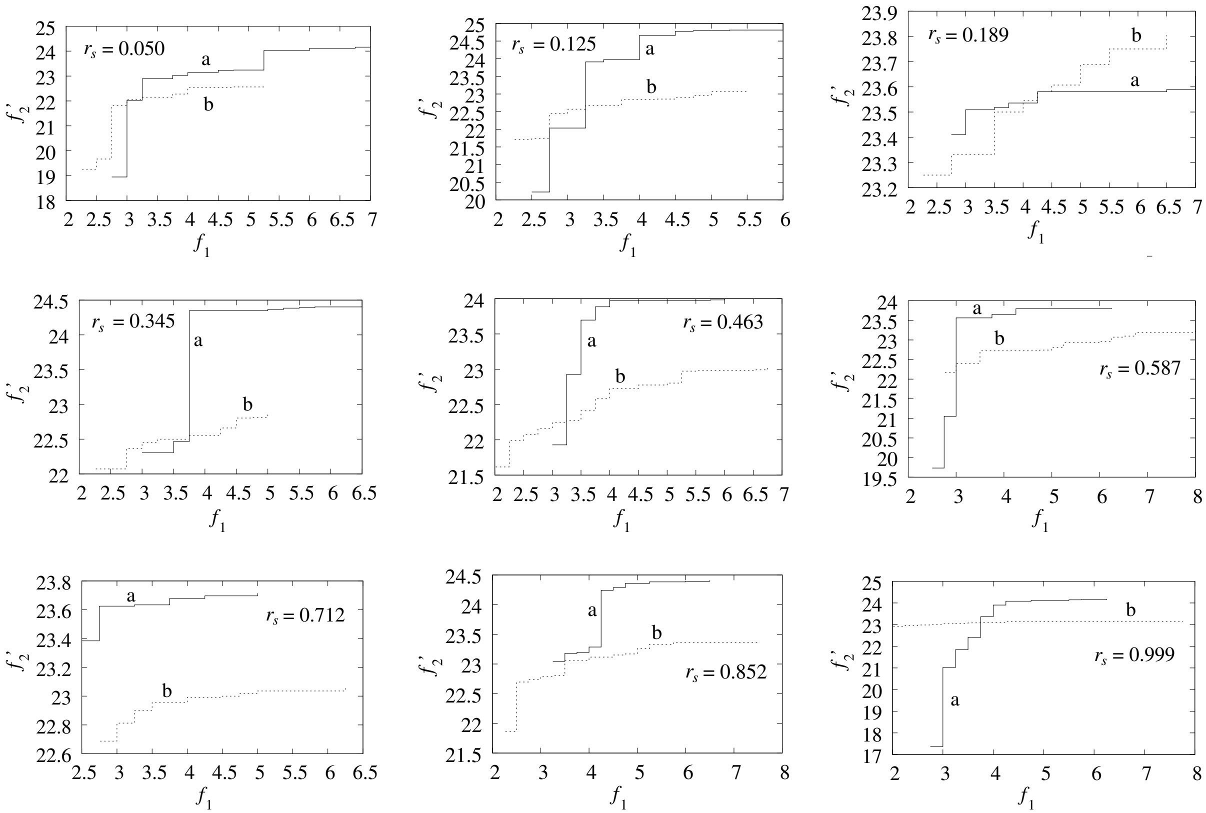


Fig. 22: Curve (a): combined XVRA and MRRA, and curve (b): MRRA alone. Combined XVRA and MRRA is found better.

Properties of NSGA-II-UCTO

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	T31	T32	T33	T34	T35
R1	64	151	148	34	28	30	45	148	84	89	64	34	1	80	45	30	70	73	136	128	68	70	58	86	30	28	92	52	68	160	117	148	45	70	92
R2	113	122	84	80	137	160	1	109	104	13	101	99	143	7	131	101	1	86	4	89	34	84	167 200	101	160	68	72	135	7	122	55	64	125	142	89
R3	177	72	117	121	167 200	4	133	76	142	151	4	121	92	52	122	77	100	113	167 200	13	80	208	133	99	156	20	55	21	135	181 207	20	140	52	32	13
R4	198	55	62	165 196	76	127	107	36	5	113	154	137	60	66		127	118	179	115	139	7	32	139	42	125	118	154	190	118	130	83	127	86	104	133
R5	170		134	131	119	81		65	33	138	184	206	185	114	190	184	178	172	223	163	170	35	184	223	187	176	123			191	220	43	141	217	170
R6	190		188	161	221	199	156	128	100	136	187	188	202					206		220		152	221		216	188	31	213		65	48	187	182	114	109
R7		176	206		56	171		176	37	217	83	24	215			14	166		215		5	56			198		217	93	152	93	47		134	138	105
R8	212		95	53		53			123	87	212	16		97	33	105	26		69	161		222		2	178	48	51	149	51		193	95	2		215
R9	19	81	94	223		222		67	69	221	173	31	116	96		87	90		85	35	174	9	29	119	102	47	50		50	102	192	94	197	174	29
R10	145	195	147	51	97	71	225	48	71	95	225	147	26	145	149	193	198	48	145	51	225			193	95	97	195	67	195		147	85	97		37
R11	144	194	146	50	96		224	47	46	94	224	146	90	144	175	192	77	47	144	50	224		116	192	94	96	194		194	46	146	19	96	14	
R12	115	168 204	20	60	42	104	21		135	117	130	58	72	165 196	165 196	121	32	40	36	107	108	141	158	151	25	59	16	172	178	15	99	172	154	59	58
R13	82		8	186 210	59	18	162 201	125	220	140	17	8	40	181 207		44	42	8	143	15	25	107	108	11	140	155	130	9	111	139	155	82	108	162 201	17
R14	23	218	181 207	15	155	44	36			18	115	186 210	82		17	168 204		23	76	186 210	142	162 201	66	168 204	43	212			18	222	137	44	66	24	
R15	3		3							54							150		54							103					103			150	
R16			153			57									6		153		57							106					106		6		
R17									63			112			159									74		12				74					
R18					91					79			79			91																			
R19		110															61			157										10					
R20			78		120					120		164				22				203						78							22		
R21		39								38										39			38												
R22					41					41																									
R23			49						49																										
R24			75																					75											
R25																														88				88	
R26			98					98																											
R27																				124				124											
R28																	126										126								
R29			129																					129											
R30																	132							132											
R31									205																	169									
R32						180							189																						
R33					209				183																										
R34																															211				
R35																							214												
R36								219																											
R37																	226																		
NR										27																	27								

Fig. 23: One class timetable of NITS1 under MRRA alone.

Properties of NSGA-II-UCTO (Contd...)

Table 6: % of classes of NITS1 scheduled in the same slots.

Type of Classes	Total No. of Classes	No. of Classes in the Same Slots	% of Classes in the Same Slots
Multi-Slot	72	56	77.78
Combined	36	28	77.78
Split	150	57	38.00
Group	62	18	29.03
Simple Single-Slot	254	90	35.43

Land-Use Management Problem

What is Land-Use management Problem ?

- It may be defined as the process of allocating different competitive land uses or activities, such as agriculture, forest, manufacturing industries, recreational activities or conservation, to different units/cells of a landscape to meet the desired objectives of land managers.
- It is an integer or mixed-integer programming problem, particularly combinatorial optimization problem.

Why Land-Use Management is Required ?

- Land and its resources have been under tremendous pressure since the very beginning of human civilization.
- The increasing pressure due to population rise, and human activities on land to meet various demands, are causing significant transformations of land for a variety of land uses - without any attention to their long term environmental impacts.
- Although humans can improve the properties of soil through their agricultural activities, by far the most common effects of human activities on soil are degradation and destruction, and environmental instability (Huston [2006]).

What are the Today's Big Threats from Mismanagement of Land ? What are Their Remedies ?

- One burning problem from environmental instability, caused by mismanagement of land, is global warming. It can be balanced only through carbon sequestration.
- Another big issue, associated with land uses, is soil degradation - the visible part of which is soil erosion (Anthoni [2006]). Hence, degradation can be reduced by reducing soil erosion.

Literature Review on Land-use Management Problem

- Very limited number of works have been done so far.
- Potentialities of optimization tools are yet to be exploited properly.
- Classical methods are not fully capable to handle the problem, and mainly evolutionary algorithms are being attempted to overcome the drawbacks of classical methods.

Aim of the Present Work on Land-use Management Problem

- Land-use management has emerged today as a problem of great concern, which needs extensive study of mechanistic models before deploying it to the real field.
- Various non-commensurable objectives of land managers can be achieved only through optimization tools.

- Hence, the present work has been aimed at formulating the problem as a multi-objective optimization problem, and developing a spatial-GIS based evolutionary algorithm for handling it.

Carbon Sequestration

- Carbon sequestration means lowering the atmospheric concentration of carbon (CO_2) and storing it in soil by one or both of the following processes:
  - Reducing its emissions through the reduction in the demand of fossil fuels, and/or other human activities, such as deforestation and land-use changes, and
  - Increasing the rates of its removal from the atmosphere through the growth of terrestrial biomass, and storing it in terrestrial level (Bhadwal and Singh [2002], USDA:GCFS [2006]).

Soil Erosion

- During soil erosion, layers of soil are transported, from one area to another, by gravity, water or wind (Anthoni [2006]).
- Although plants can provide protective cover on land, the loss of protective plants makes soil vulnerable for being eroded.
- Moreover, over-cultivation and compaction cause soil to lose its structure and cohesion, thus becoming more easily erodible.
- Besides water pollution, soil erosion sweeps away soil carbon and nutrient-rich productive layer of *humus* or *topsoil*.
- Amount of soil eroded from a unit can be estimated by the following *Universal Soil Loss Equation* (USLE) (McCloy [1995], RUSLE [2006]):

$$\epsilon = R.K.LS.C.P \quad . \tag{13}$$

Constraints in Land-use Management Problem

- Physical constraints on geomorphological structure:
  1. A land-use should be applied in a unit, only if it is permitted in the soil of that unit,
  2. The slope of a unit should be within the permitted range of slope for the land-use applied in that unit,
  3. The aridity index of a unit should be within the permitted range of aridity index for the land-use applied in that unit.
  4. *Topographic Soil Wetness Index* (TSWI) of a unit should be within the permitted range of TSWI for the land-use applied in that unit.

Constraints in Land-use Management Problem (Contd...)

- Ecological constraints on spatial coherence:
  1. Area in a patch of a land-use should be within the permitted range of area in a patch for that land-use, and
  2. Total area under a land-use in a landscape should be within the permitted range of total area for that land-use.

Objectives in Land-use Management Problem

1. Maximize net present economic return,
2. Maximize net amount of carbon sequestration, and
3. Minimize net amount of soil erosion.

Mathematical Formulation

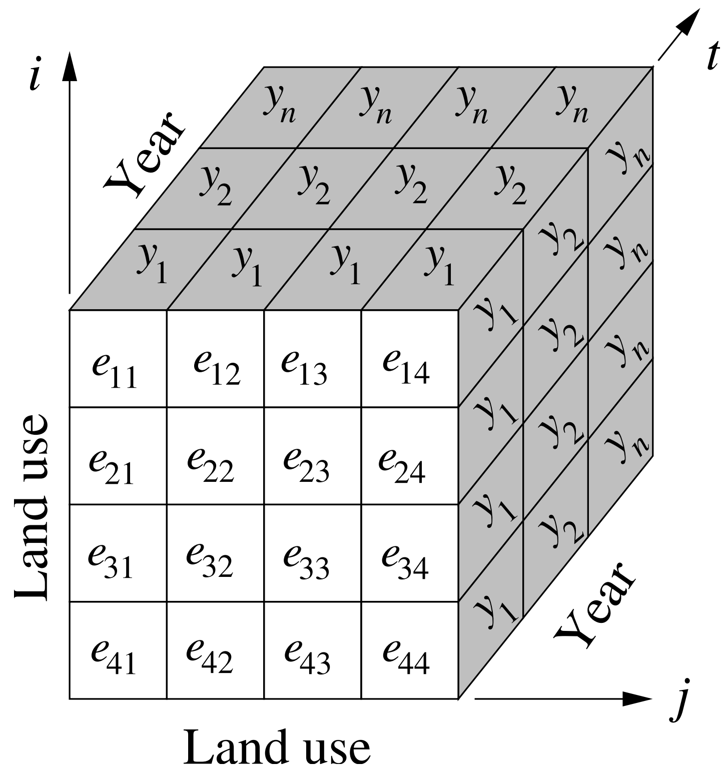


Fig. 24: Representation of a landscape.

Mathematical Formulation (Contd...)

- Objective functions:
  1. Maximize net present economic return

$$f_1 \equiv \sum_{e=1}^{\mathrm{E}} \sum_{i=1}^{\mathrm{R_u}} \sum_{j=1}^{\mathrm{C_u}} X_{e,i,j}.m_{e,i,j} \quad , \tag{14}$$

  2. Maximize net amount of carbon sequestered

$$f_2 \equiv \sum_{e=1}^{\mathrm{E}} \sum_{i=1}^{\mathrm{R_u}} \sum_{j=1}^{\mathrm{C_u}} \sum_{t=1}^{\mathrm{T}} X_{e,i,j}.C_{e,i,j,t} \quad , \tag{15}$$

  3. Minimize net amount of soil eroded

$$f_3 \equiv \sum_{e=1}^{\mathrm{E}} \sum_{i=1}^{\mathrm{R_u}} \sum_{j=1}^{\mathrm{C_u}} \sum_{t=1}^{\mathrm{T}} X_{e,i,j}.\epsilon_{e,i,j,t} \quad , \tag{16}$$

Mathematical Formulation (Contd...)

- Subject to physical constraints on geomorphological structure:

$$e^{\mathrm{u}}_{i,j} \in \mathrm{E}^{\mathrm{u}}_{i,j}, \quad i = 1, .., \mathrm{R_u} \text{ and } j = 1, .., \mathrm{C_u} \quad , \tag{17}$$

1. Type of soil of a unit

$$s_{i,j} \in S_e \quad , \tag{18a}$$

2. Slope of a unit

$$L^{\min}_{e,s_{i,j}} \leq l_{i,j} \leq L^{\max}_{e,s_{i,j}} \quad , \tag{18b}$$

3. Aridity index (P/PET) of a unit

$$D^{\min}_{e,s_{i,j}} \leq d_{i,j} \leq D^{\max}_{e,s_{i,j}} \quad , \tag{18c}$$

4. Topographic soil wetness index (TSWI) of a unit

$$H^{\min}_{e,s_{i,j}} \leq h_{i,j} \leq H^{\max}_{e,s_{i,j}} \quad , \tag{18d}$$

Mathematical Formulation (Contd...)

- Ecological constraints on spatial coherence:
  1. Area in a patch of a land-use

$$\left.\begin{aligned} g_{2(\sum_{e'=0}^{e-1} N_{e'}+n)-1} &\equiv a_{e,n} \geq A^{\mathrm{p}}_{e,\mathrm{min}} \\ g_{2(\sum_{e'=0}^{e-1} N_{e'}+n)} &\equiv a_{e,n} \leq A^{\mathrm{p}}_{e,\mathrm{max}} \end{aligned}\right\}, e = 1,..,\mathrm{E};\ n = 1,..,N_e;\ \ \&\ N_0 = 0, \quad (19)$$

  2. Total area under a land-use

$$\left.\begin{aligned} g_{2(\sum_{e'=1}^{\mathrm{E}} N_{e'}+e)-1} &\equiv \sum_{n=1}^{N_e} a_{e,n} \geq A_{e,\mathrm{min}} \\ g_{2(\sum_{e'=1}^{\mathrm{E}} N_{e'}+e)} &\equiv \sum_{n=1}^{N_e} a_{e,n} \leq A_{e,\mathrm{max}} \end{aligned}\right\}, \quad e = 1,..,\mathrm{E}\ , \quad (20)$$

Mathematical Formulation (Contd...)

1. Number of objective functions : 3
2. Number of physical constraints : U
3. Number of ecological constraints : $(2\sum_{e=1}^{\mathrm{E}} N_e + \mathrm{E})$

where,

U	=	Number of units in a landscape,
N_e	=	Number of patches under event e,
E	=	Total number of evenets (land uses).

Chromosome Representation

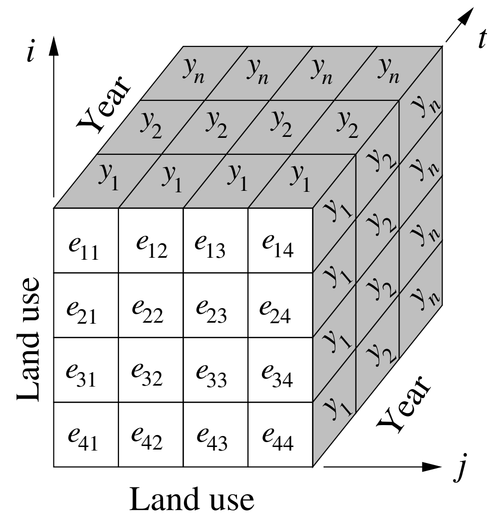


Fig. 25: Representation of a landscape.

$$\left.\begin{aligned} \mathbf{G} &= [e_{ij}]_{\mathrm{R_u x C_u}} \\ e_{ij} &= (y_1, y_2, ..., y_t, ..., y_\mathrm{T})^T \end{aligned}\right\} \quad (21)$$

Two-Dimensional Crossover Operator (XTD)

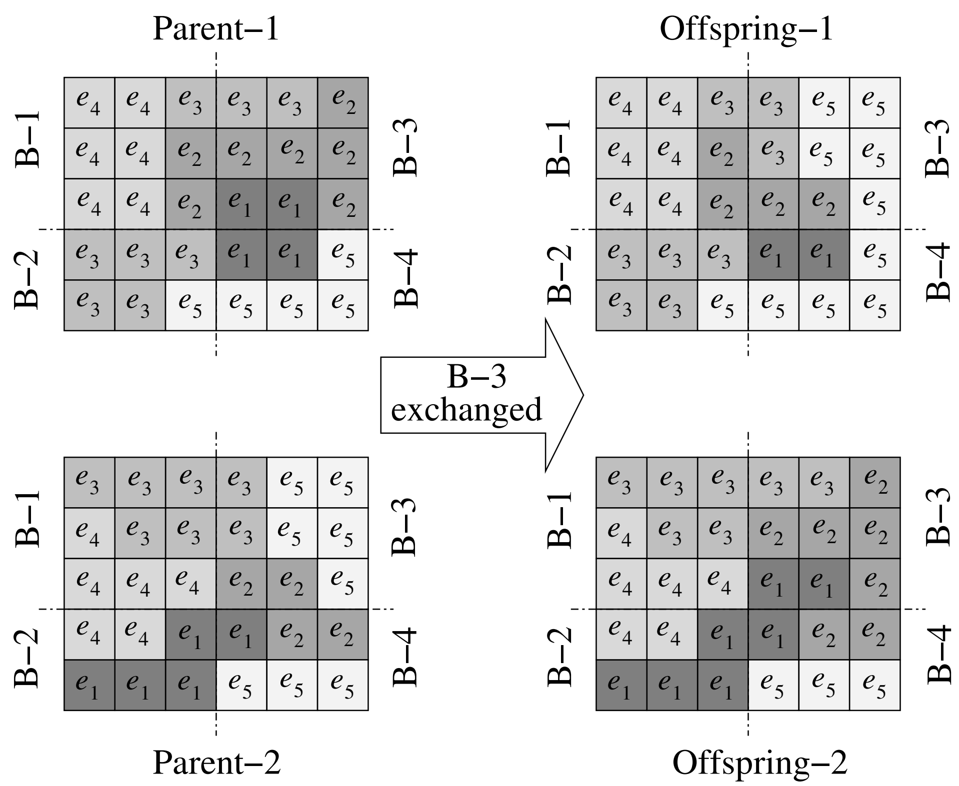


Fig. 26: XTD in land-use management problem.

Crossover on Boundary Cells (XBC)

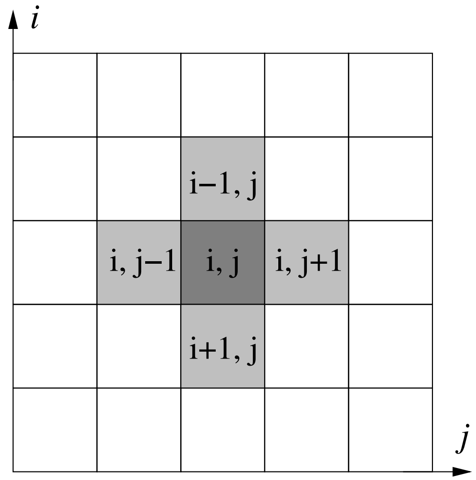


Fig. 27: Adjacent cells of cell (i,j).

- Find the Hamming distance of two paprents.
- If a Hamming cell of a parent is on boundary, replace its land-use by that of the corresponding cell of the second parent.

Mutation Operators (MBC & MSIS2)

- Mutation on Boundary Cells (MBC)
  - Replace the land-use of a random boundary cell by the one in one of its adjacent cells, having different land-use than in the chosen boundary cell.
- Mutation for Steering Infeasible Solution-2 (MSIS2).
  - If the area of the patch, in which a random boundary cell belongs, is less than its minimum requirement, merge in the patch the adjacent cells of the chosen cell.

Guidance for Satisfying Spatial Requirements

1. During initialization, an attempt may be made for satisfying, if possible, the patch-size constraints of a land-use by scheduling it in the sufficient number of contiguous/adjacent units.

2. During optimization process, a patch in a solution, having less area than the specified one, may be deleted before evaluating the solution. This can be made by merging the cells of a patch in its adjacent patches.

NSGA-II-LUM: NSGA-II in Land-Use Management

- The proposed chromosome representation and EA operators are incorporated in NSGA-II to handle land-use management problem.
- Local search (Deb and Goel [2001]), as given below, may also be incorporated with NSGA-II-LUM to enhance the final Pareto front.

$$\text{Minimize } F(\boldsymbol{x}) \equiv \sum_{i=1}^{\mathcal{M}} \bar{w}_i^{\boldsymbol{x}} f_i(\boldsymbol{x}) \ , \tag{22}$$

where $\mathcal{M}$ is the number of objective functions, and $\bar{w}_i^{\boldsymbol{x}}$ is pseudo-weight for i-th objective function of solution $\boldsymbol{x}$. $\bar{w}_i^{\boldsymbol{x}}$ can be calculated as:

$$\bar{w}_i^{\boldsymbol{x}} = \frac{(f_i^{\max} - f_i(\boldsymbol{x}))/(f_i^{\max} - f_i^{\min})}{\sum_{j=1}^{\mathcal{M}}((f_j^{\max} - f_j(\boldsymbol{x}))/(f_j^{\max} - f_j^{\min}))} \ , \tag{23}$$

Application of NSGA-II-LUM

- NSGA-II-LUM has been applied to a Mediterranean landscape, located in Baixo Alentejo ($38^0 0' 50.3''$ N and $7^0 51' 56.94''$ W), Southern Portugal. It has been named here as LBAP in short.
  - The landscape is divided into 100×100 units.
  - Five different land uses are permitted in the landscape
    1. Annual agriculture,
    2. Permanent agriculture,
    3. Mixed agriculture,
    4. Forest, and
    5. Shrubs.

LBAP Without Guidance to Patch-Size Constraints

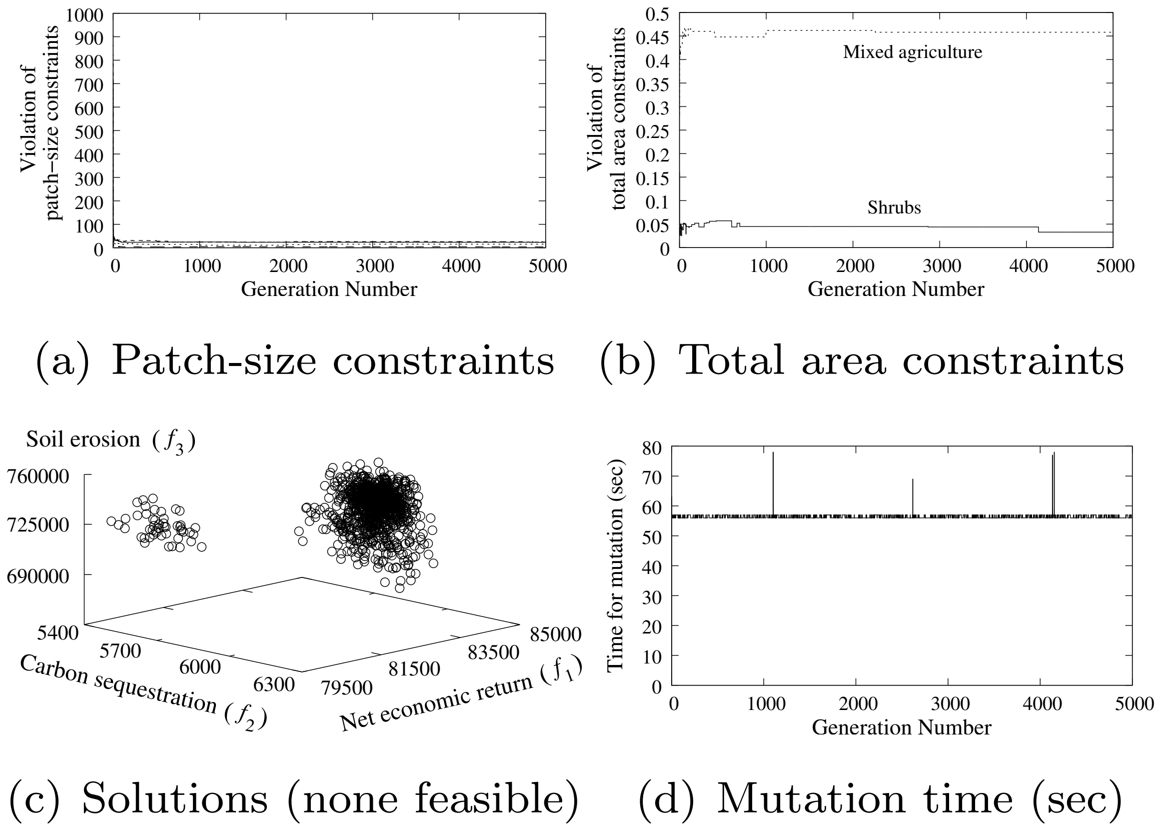
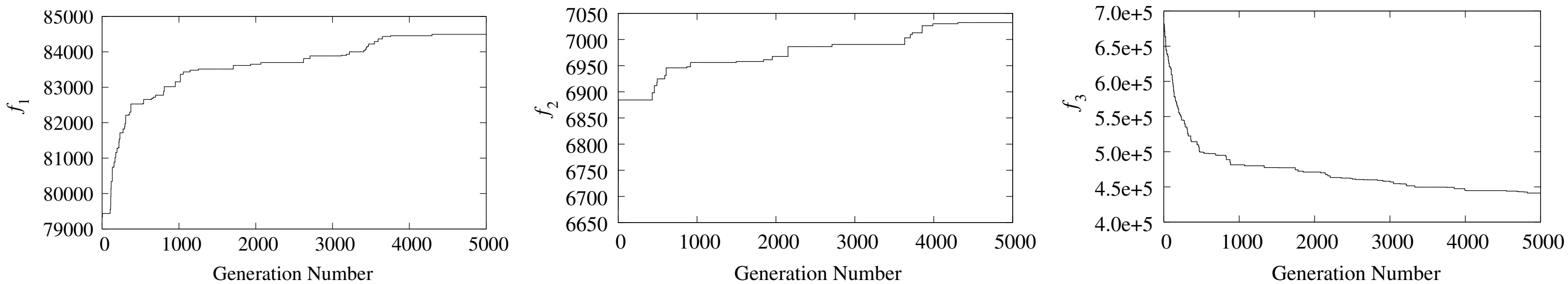
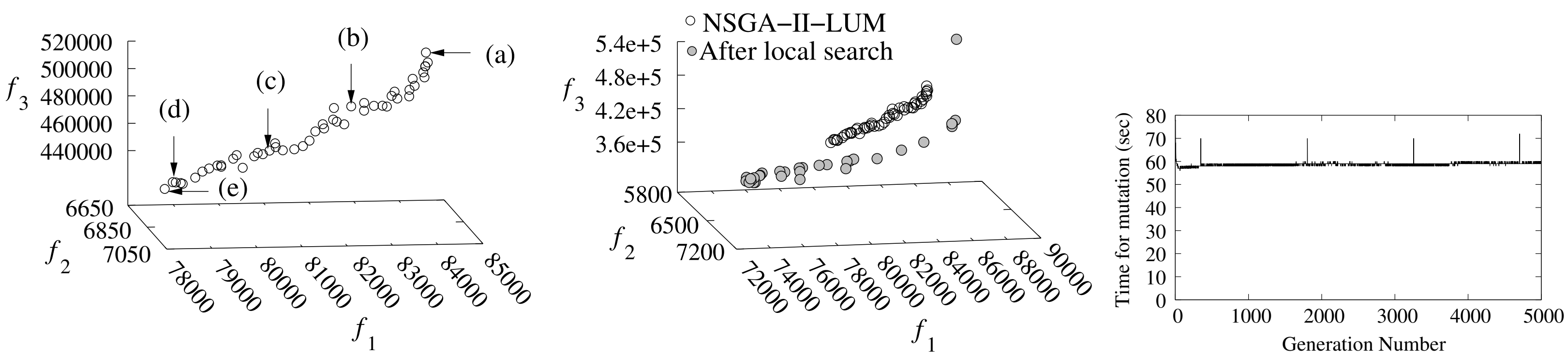


Fig. 28: LBAP without guidance for patch-size constraints.

LBAP Using XTD



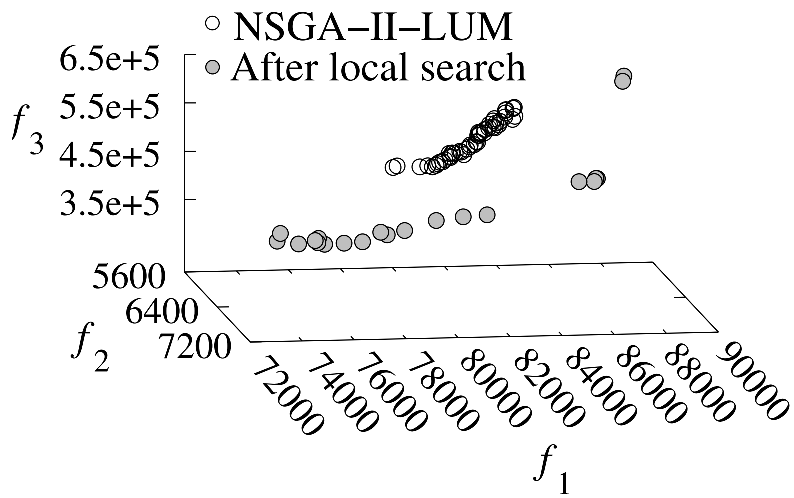
(a) Improvement of f_1 (b) Improvement of f_2 (c) Improvement of f_3



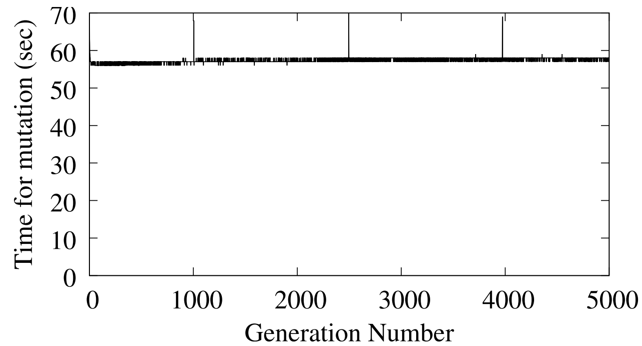
(d) NSGA-II-LUM (e) NSGA-II-LUM & LS (f) Mutation time

Fig. 29: Solution of LBAP using XTD.

LBAP Using XBC



(a) NSGA-II-LUM & LS



(b) Mutation time

Fig. 30: Solution of LBAP using XBC.

Execution time for 5000 generation = 80 hours 12 minutes 10 seconds.

Comparison of Results of LBAP from XTD and XBC

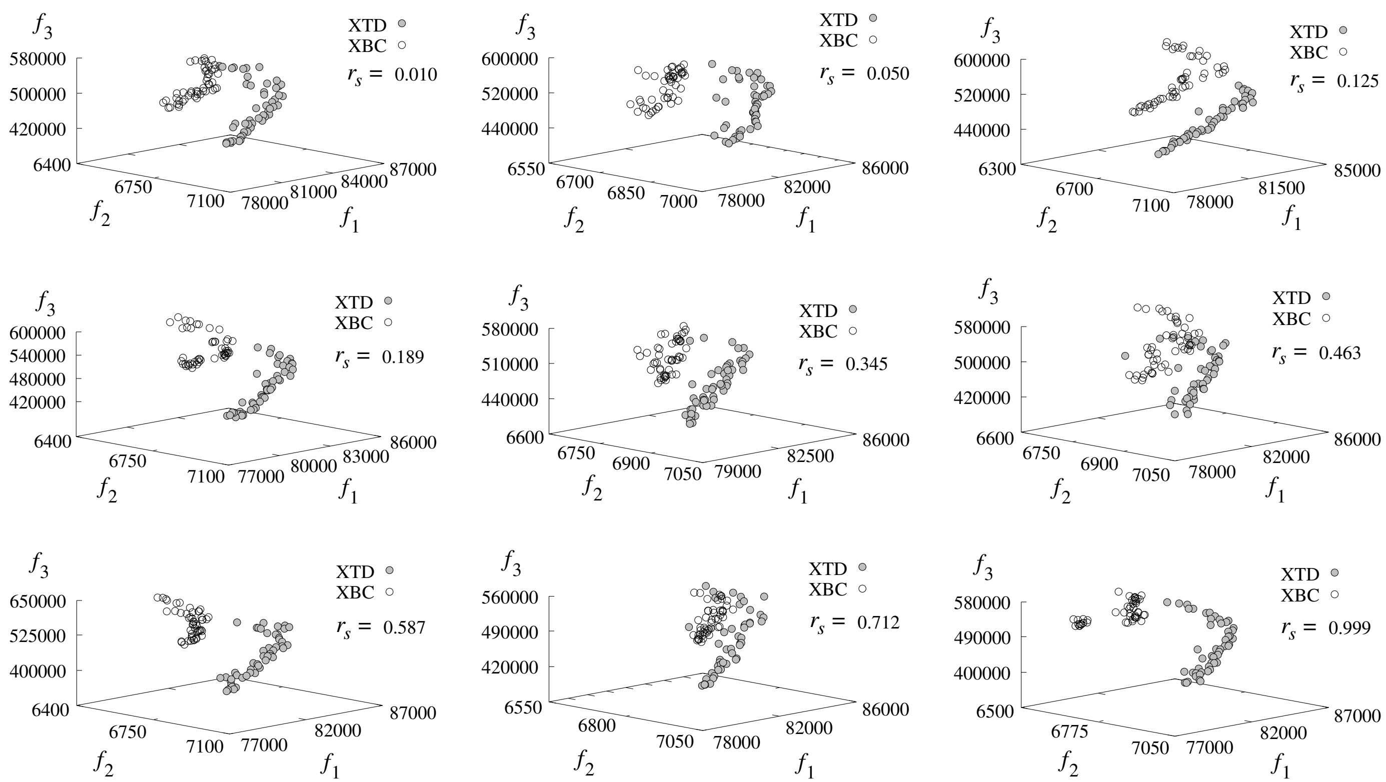


Fig. 31: Comparison of Pareto fronts of LBAP, obtained using XTD and XBC. In all cases, XTD has outperformed XBC.

Distribution of Land Uses in LBAP by NSGA-II-LUM

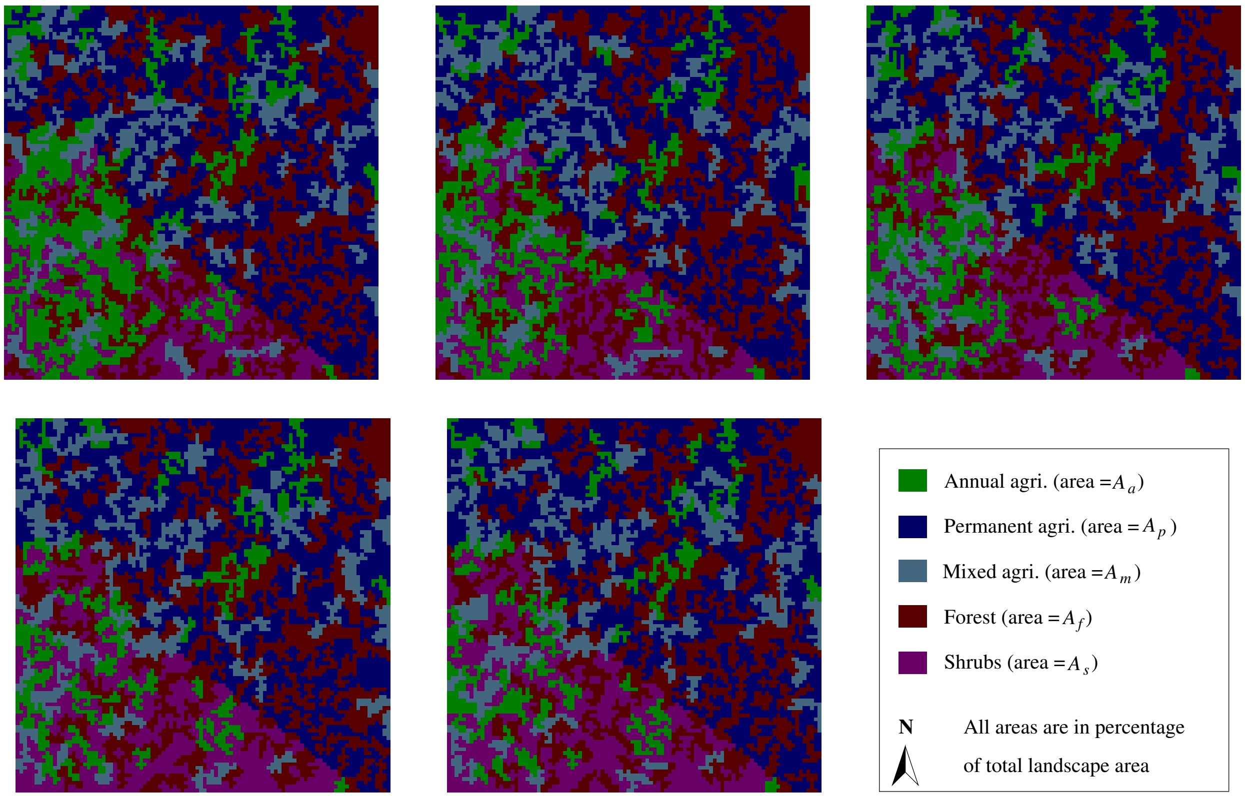
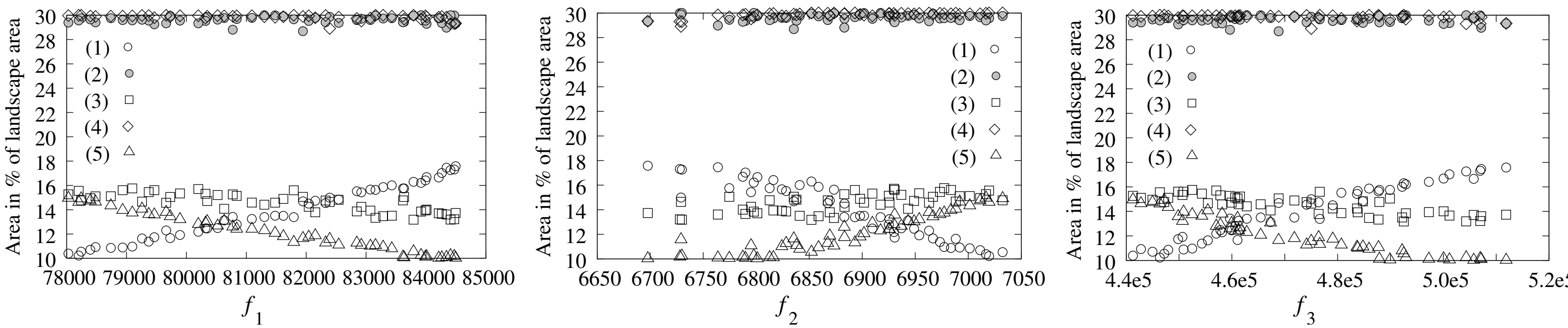


Fig. 32: Distribution of land uses in LBAP by NSGA-II-LUM.

Area Distribution in LBAP



(a) f_1 vs area (b) f_2 vs area (c) f_3 vs area

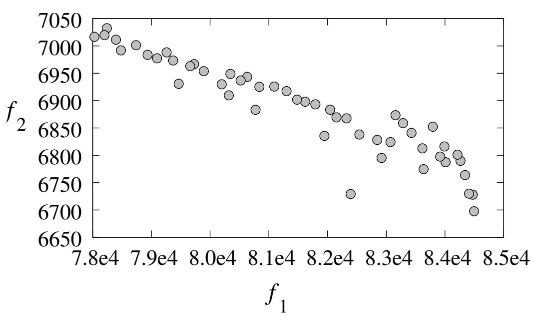
Fig. 33: Objective-wise distribution of areas under different land uses of LBAP. (1): Annual agriculture, (2): Permanent agriculture, (3): Mixed agriculture, (4): Forest, and (5): Shrubs.

Variation of Areas in LBAP

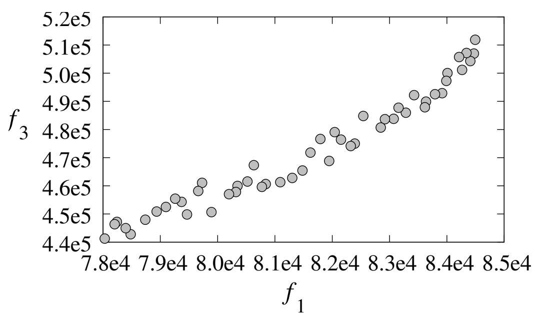
Table 7: Objective-wise variation of areas in LBAP.

Objectives		**Land Uses of LBAP**				
		Annual Agri.	**Permanent Agri.**	**Mixed Agri.**	**Forest**	**Shrubs**
Economic (f_1)	↑	↑	Maximum possible value	↓	Maximum possible value	↓
Carbon (f_2)	↑	↓	Maximum possible value	↑	Maximum possible value	↑
Soil (f_3)	↓	↓	Maximum possible value	↑	Maximum possible value	↑

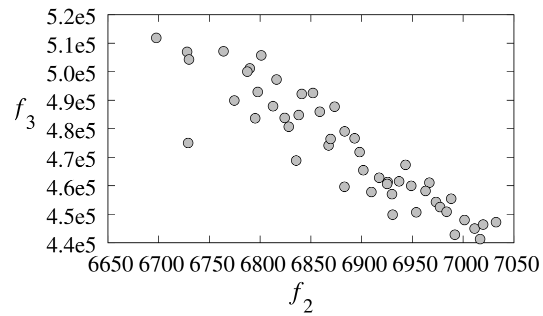
Natures of the Objective Functions of LBAP



(a) f_1 vs f_2



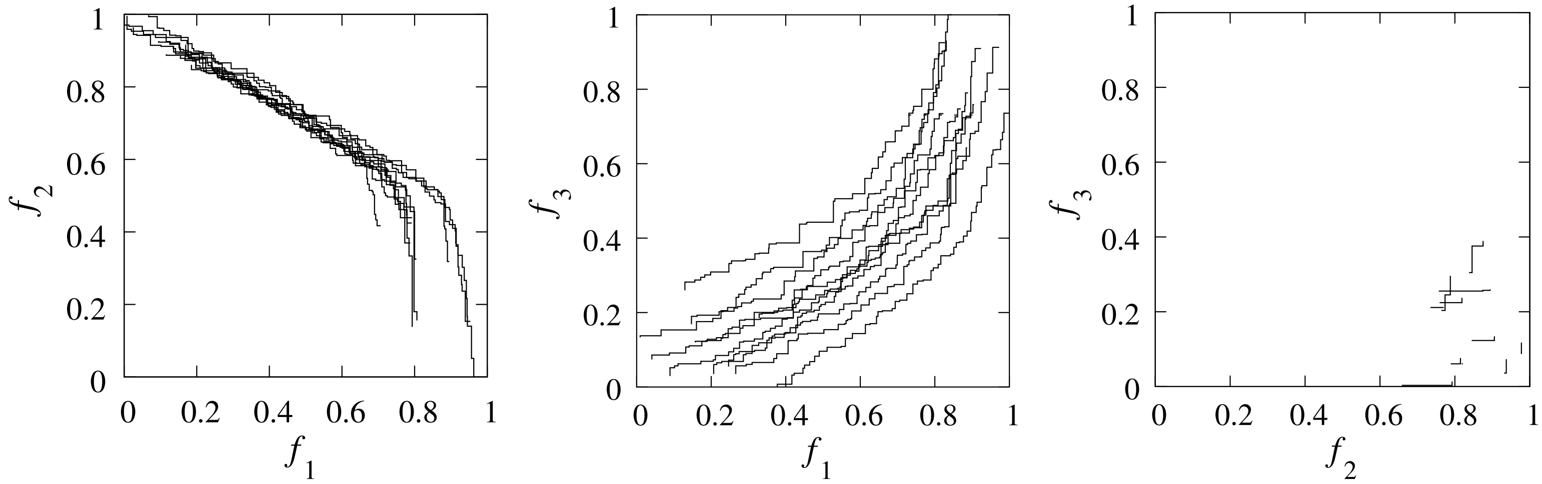
(b) f_1 vs f_3



(c) f_2 vs f_3

Fig. 34: 2D projections of the objective values of a Pareto front of LBAP. f_1 is conflicting with both f_2 and f_3, but f_2 and f_3 are correlated.

Two-Objective Optimization of LBAP



(a) Objectives: f_1 and f_2 (b) Objectives: f_1 and f_3 (c) Objectives: f_2 and f_3

Fig. 35: Attainment surfaces of LBAP under two-objective optimization. f_1 is conflicting with both f_2 and f_3, but f_2 and f_3 are correlated.

- NSGA-II-LUM is dependent on initial solutions.

Similarities Between Class Timetabling and Land-Use Management Problems

1. Both are spatial and temporal based problems.
2. Both the problems require multiple objectives to be met simultaneously
3. Both are highly constrained combinatorial optimization problems
4. In class timetabling problem, each event (class) is required to be scheduled exactly once, while an event (land-use) in land-use management problem may be scheduled multiple times within a certain range. These requirements slightly differ the problems from one another.

Similarities Between Their Solution Techniques

1. Classical algorithms are not fully capable to handle the problems.
2. Both of NSGA-II-UCTO and NSGA-II-LUM need some guidance to initialize the solutions in their populations.
3. Two-dimensional matrix is applicable for solution representations for both the problems.
4. Two dimensional crossover operators can be used for both the problems.
5. Constraint-structures and other problem information may be exploited in developing EA operators.

Conclusions

- NSGA-II-UCTO, a multi-objective EA-based optimizer, has been developed for handling university class timetabling problem.
  1. It is directly applicable to university class timetabling. It can also be applied to school timetabling with a little modification,
  2. Different types of classes, such as multi-slot, split, combined and group classes, can be handled through input datafiles only,
  3. Choices for rooms and time-slots can also be made through input datafiles only,
  4. Moreover, status of constraint handling can also be made through input datafiles, and
  5. Addition/deletion of any constraint can be made through subroutines.

Conclusions (Contd...)

NSGA-II-UCTO has been applied successfully to three real problems from two technical institutes in India. The findings from its application are:

1. It is able to maintain good trade-off between the objective functions.
2. However, it is dependent on user-defined mutation probabilities - which can be sorted out to some extent by using evolving mutation probabilities.
3. It is dependent on initial solutions also - for which multiple runs may be performed with different initial solutions, and then the best solution can be captured from different alternatives.
4. Out of the considered three problems, it has performed well on one with mutation operator alone, while on other two with combined crossover and mutation operators. Hence, a problem may be tested under both the cases to have the best alternative.
5. It has the tendency for allocating a particular class in a particular slot of a timetable.

Conclusions (Contd...)

- NSGA-II-LUM, another multi-objective EA, based on spatial-GIS configuration, has been developed for land-use management problem. It has been applied to a Mediterranean landscape located in Southern Portugal, and observed that
  1. It is able to maintain good trade-off among the objective functions. Moreover, it has the tendency for allocating a particular land-use in a particular location of a landscape.
  2. Both carbon sequestration and soil erosion conflict with economic return. However, carbon sequestration and soil erosion are correlated to some extent.
  3. Higher economic return prefers higher annual agriculture, and lower mixed agriculture and shrubs. The preferences of higher carbon sequestration and lower soil erosion are just opposite to those of higher economic return.
  4. Permanent agriculture and forest are preferred to the maximum extent by all three objective functions, irrespective to their values.

Conclusions (Contd...)

- During the process of studying the above two problems, and findings therewith, a number of similarities between the problems, and also between their solution techniques, have been encountered.
    1. Both are highly constrained, and spatial and temporal-based, multi-objective combinatorial optimization problems. Class timetabling problem requires each event (class) to be scheduled exactly once, while an event (land-use) in land-use management problem may be scheduled multiple times within a certain range - which bring a slight difference in the solution techniques for these two problems.
    2. Classical methods are not fully capable to handle any of the problems. If solved by EAs, similar chromosome/solution representations and EA operators, based on individual problem-information, can be used in both the problem.

Future Research Scope

1. The heuristic approach, developed for generating initial solutions to university class timetabling problem, is computationally expensive, particularly when a problem has limited number of free slots. Attempt may be made to develop some faster approach.

2. Though similar in natures, exactly the same EA operators can not be used in both the problems, but they need to be problem specific (except XTD in land-use management problem). These can be made more general, if dependency on problem-information can be reduced.

3. NSGA-II-UCTO and NSGA-II-LUM are not guaranteed to come out of infeasible regions. Hence, some guidance are required either to generate feasible solutions or get rid of infeasible solutions.

4. Though an experiment has been made for finding the effective probability of Hamming cells, which can be used in XBC for generating good solutions for land-use management problem, the attempt was not successful.

5. The EAs, developed in the present work, are dependent on initial solutions and user-defined mutation probabilities.

References

Anthoni, J. F. (2006, January). *Soil: erosion and conservation.* `http://www.seafriends.org.nz/enviro/soil/erosion.htm`.

Bhadwal, S. and Singh, R. (2002). Carbon sequestration estimates for forestry options under different land-use scenarios in India. *Current Science 83*(11), 1380–1386.

Deb, K. and Goel, T. (2001). A hybrid multi-objective evolutionary approach to engineering shape design. In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization (EMO-2001), Zürich, Switzerland*, pp. 385–399.

Huston, M. (2006, January). *The need for science and technology in land management.* Online Book - The International Development Research Centre, `http://www.idrc.ca/en/ev-29587-201-1-DO_TOPIC.html`.

Ibaraki, T. and Katoh, N. (1988). *Resource allocation problems: Algorithmic approaches.* MIT Press Series in the Foundations of Computing.

McCloy, K. R. (1995). *Resource Management Information Systems: Process and Practice.* Taylor & Francis Ltd.

RUSLE (2006, January). *Online soil erosion assessment tool - RUSLE*. Institute of Water Research, Michigan State University, `http://www.iwr.msu.edu/rusle`.

USDA:GCFS (2006, January). *USDA (United States Department of Agriculture) Global change fact sheet - Soil carbon sequestration: Frequently asked questions*. `http://www.usda.gov/oce/gcpo/sequeste.txt`.

Zhang, W. (2002). Modeling and solving a resource allocation problem with soft constraint techniques. Technical Report WUCS-2002-13, Department of Computer Science and Engineering, Washington University in St. Louis.

THANK YOU !