C interfaces to GALAHAD LHS

Jari Fowkes and Nick Gould
STFC Rutherford Appleton Laboratory
Tue May 2 2023

Chapter 1

GALAHAD C package lhs

1.1 Introduction

1.1.1 Purpose

This package **computes an array of Latin Hypercube samples.**.

Currently, only the control and inform parameters are exposed; these are provided and used by other GALAHAD packages with C interfaces.

1.1.2 Authors

J. Burkardt, University of Pittsburgh (LGPL) adapted for GALAHAD by N. I. M. Gould, STFC-Rutherford Appleton Laboratory, England.

C interface, additionally J. Fowkes, STFC-Rutherford Appleton Laboratory.

Julia interface, additionally A. Montoison and D. Orban, Polytechnique Montréal.

1.1.3 Originally released

June 2016, C interface March 2022.

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

File Documentation

3.1 galahad_lhs.h File Reference

```
#include <stdbool.h>
#include <stdint.h>
#include "galahad_precision.h"
#include "galahad_cfunctions.h"
```

Data Structures

- struct lhs_control_type
- struct lhs_inform_type

Functions

- void lhs_initialize (void **data, struct lhs_control_type *control, struct lhs_inform_type *inform)
- void lhs_read_specfile (struct lhs_control_type *control, const char specfile[ ])
- void lhs_ihs (int n_dimen, int n_points, int *seed, int **X, const struct lhs_control_type *control, struct lhs_inform_type *inform, void **data)
- void lhs_get_seed (int *seed)
- void lhs_terminate (void **data, struct lhs_control_type *control, struct lhs_inform_type *inform)

3.1.1 Data Structure Documentation

3.1.1.1 struct lhs_control_type

Data Fields

int	error	error and warning diagnostics occur on stream error.
int	out	general output occurs on stream out.
int	print_level	the level of output required. Possible values are: • $<$ 1 no output. • $>$ 0 debugging.

Data Fields

int	duplication	the duplication factor. This must be at least 1, a value of 5 is reasonable.
bool	space_critical	if .space_critical true, every effort will be made to use as little space as possible. This may result in longer computation time.
bool	deallocate_error_fatal	if .deallocate_error_fatal is true, any array/pointer deallocation error will terminate execution. Otherwise, computation will continue.
char	prefix[31]	all output lines will be prefixed by .prefix(2:LEN(TRIM(prefix))-1) where .prefix contains the required string enclosed in quotes, e.g. "string" or 'string'

3.1.1.2 struct lhs_inform_type

Data Fields

int	status	return status. Possible values are: • 0 the call was successful. • -1. An allocation error occurred. A message indicating the offending array is written on unit control.error, and the returned allocation status and a string containing the name of the offending array are held in inform.alloc_status and inform.bad_alloc respectively. • -2. A deallocation error occurred. A message indicating the offending array is written on unit control.error and the returned allocation status and a string containing the name of the offending array are held in inform.alloc_status and inform.bad_alloc respectively. • -3. The random number seed has not been set.
int	alloc_status	the status of the last attempted allocation/deallocation.
char	bad_alloc[81]	the name of the array for which an allocation/deallocation error occurred.

3.1.2 Function Documentation

3.1.2.1 lhs_initialize()

```
void lhs_initialize (
             void ** data,
             struct lhs_control_type * control,
             struct lhs_inform_type * inform )
```

3.1.2.2 lhs_read_specfile()

```
void lhs_read_specfile (
             struct lhs_control_type * control,
             const char specfile[] )
```

3.1.2.3 lhs_ihs()

```
void lhs_ihs (
             int n_dimen,
             int n_points,
             int * seed,
             int ** X,
             const struct lhs_control_type * control,
             struct lhs_inform_type * inform,
             void ** data )
```

3.1.2.4 lhs_get_seed()

```
void lhs_get_seed (
             int * seed )
```

3.1.2.5 lhs_terminate()

```
void lhs_terminate (
             void ** data,
             struct lhs_control_type * control,
             struct lhs_inform_type * inform )
```